
When Kernels Multiply, Clusters Unify: Fusing Embeddings with the Kronecker Product

Youqi Wu*

yqwu24@cse.cuhk.edu.hk

Jingwei Zhang*

jwzhang22@cse.cuhk.edu.hk

Farzan Farnia*

farnia@cse.cuhk.edu.hk

Abstract

State-of-the-art embeddings often capture distinct yet complementary discriminative features: For instance, one image embedding model may excel at distinguishing fine-grained textures, while another focuses on object-level structure. Motivated by this observation, we propose a principled approach to fuse such complementary representations through *kernel multiplication*. Multiplying the kernel similarity functions of two embeddings allows their discriminative structures to interact, producing a fused representation whose kernel encodes the union of the clusters identified by each parent embedding. This formulation also provides a natural way to construct *joint kernels* for paired multi-modal data (e.g., image–text tuples), where the product of modality-specific kernels inherits structure from both domains. We highlight that this kernel product is mathematically realized via the *Kronecker product* of the embedding feature maps, yielding our proposed *KrossFuse* framework for embedding fusion. To address the computational cost of the resulting high-dimensional Kronecker space, we further develop *RP-KrossFuse*, a scalable variant that leverages random projections for efficient approximation. As a key application, we use this framework to bridge the performance gap between cross-modal embeddings (e.g., CLIP, BLIP) and unimodal experts (e.g., DINOv2, E5). Experiments show that RP-KrossFuse effectively integrates these models, enhancing modality-specific performance while preserving cross-modal alignment. The project code is available at <https://github.com/yokiwuuu/KrossFuse>.

1 Introduction

The representation learning literature features a wide range of embedding models, each excelling at distinct and often complementary discriminative features. For example, one image encoder may specialize in distinguishing fine-grained categories such as dog breeds, while another captures broader semantic distinctions such as traffic signs. This contrast is illustrated in Figure 1: the DINOv2 image embedding [53] tends to form clearer clusters for dog breeds but shows less separation among traffic signs, whereas the CLIP image embedding [60] exhibits the opposite trend, achieving better separation for traffic signs while mixing the two dog breeds in the kernel heatmaps. Such observations motivate the following question: *how can we systematically fuse multiple embeddings to obtain a single representation that combines the discriminative strengths of all its parent embeddings?*

In this work, we view each embedding as inducing a *kernel similarity function* that assigns a similarity score to every pair of samples. Evaluating this function over a reference dataset produces a *kernel similarity structure*—a matrix that reflects how the embedding perceives relationships among samples

*Department of Computer Science & Engineering, The Chinese University of Hong Kong

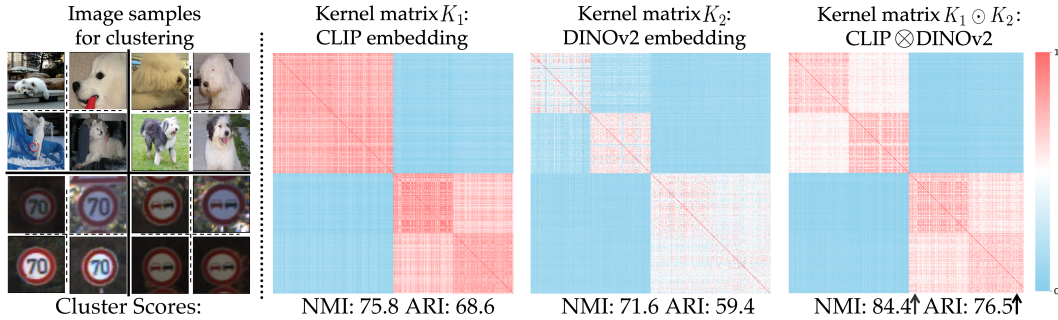


Figure 1: Heatmaps of RBF kernel similarity matrices for an image dataset with four groundtruth clusters (two dog classes in ImageNet and two traffic sign classes in GTSRB) (left) K_1 for CLIP, (middle) K_2 for DINOv2, (right) $K_1 \odot K_2$ elementwise product for CLIP and DINOv2’s Kronecker product. Unlike CLIP and DINOv2, their Kronecker product could cluster the four image classes.

and which samples it tends to cluster together. This kernel-based view allows us to analyze and combine embeddings directly in the kernel space, where our goal is to construct a fused representation whose similarity structure reflects a *strict union* of the parent cluster structures: two samples should appear similar only if *all* parent embeddings agree that they are similar.

A natural and principled way to capture this “all must agree” logic is through *kernel multiplication*, defining the fused kernel $k_{\psi_{\text{fuse}}}$ as the product of the individual kernel similarity functions k_{ψ_1} and k_{ψ_2} of embeddings ψ_1 and ψ_2 :

$$k_{\psi_{\text{fuse}}}(x, y) = k_{\psi_1}(x, y) \cdot k_{\psi_2}(x, y). \quad (1)$$

This formulation provides a simple yet effective mechanism for combining the discriminative patterns of multiple embeddings: assuming normalized kernel similarity scores bounded by 1, the fused similarity becomes small whenever any parent embedding separates the two samples². The empirical effect of this operation is illustrated in Figure 1 (right), where multiplying the CLIP and DINOv2 kernels clearly separates all four classes that each model alone fails to isolate.

A feature representation whose kernel similarity function equals the product of the individual kernels is obtained by taking the *Kronecker product* of the individual embedding feature maps, which is well known in the kernel methods literature [65, 7]. Building on this insight, we introduce *KrossFuse*, a general framework for embedding fusion based on Kronecker-product embeddings. Unlike simple concatenation of embedding feature vectors, which corresponds to the summation of individual kernels and lacks the discussed cluster unification property, KrossFuse yields a representation whose kernel unifies sample clusters across parent embeddings’ similarities as displayed in Figures 1,2.

Beyond fusing representations within a single modality, KrossFuse also offers a principled way to construct *joint kernels for multi-modal data* (e.g., image–text tuples). Such joint kernels are applicable to generative and retrieval systems (e.g., text-to-image models) that rely on paired data for training and evaluation, where the product of modality-specific kernels captures the joint structure across domains. Figure 2 illustrates this effect in a text-to-image generation setting with SD-XL Turbo model [64]: while the kernel structures of the DINOv2 image, CLIP text, and their concatenated feature vectors fail to clearly separate the six clusters defined by three professions (firefighter, chef, police officer) and two genders (male, female), their *Hadamard product*—using the kernel of the Kronecker-fused embeddings—can distinctly separate all six groups.

While conceptually elegant, the Kronecker formulation incurs a significant computational cost: the dimensionality of the fused embedding equals the product of the input dimensions (for instance, fusing 512-dimensional CLIP and 768-dimensional DINOv2 embeddings yields a 393,216-dimensional vector). To address this scalability barrier, we propose *RP-KrossFuse*, a random-projection-based extension that efficiently approximates the Kronecker feature space while preserving kernel similarities. This approach retains the theoretical properties of KrossFuse while making it practical for large-scale, high-dimensional embeddings.

²For non-negative kernels such as the Gaussian RBF or normalized even-degree polynomial kernels, the near-zero kernel similarity score directly implies separation, whereas for kernels that can take negative values (e.g., linear or cosine similarity kernels), the fusion operates through orthogonality—if two samples are nearly orthogonal in any parent space, their fused similarity remains near zero, thus preserving separation.

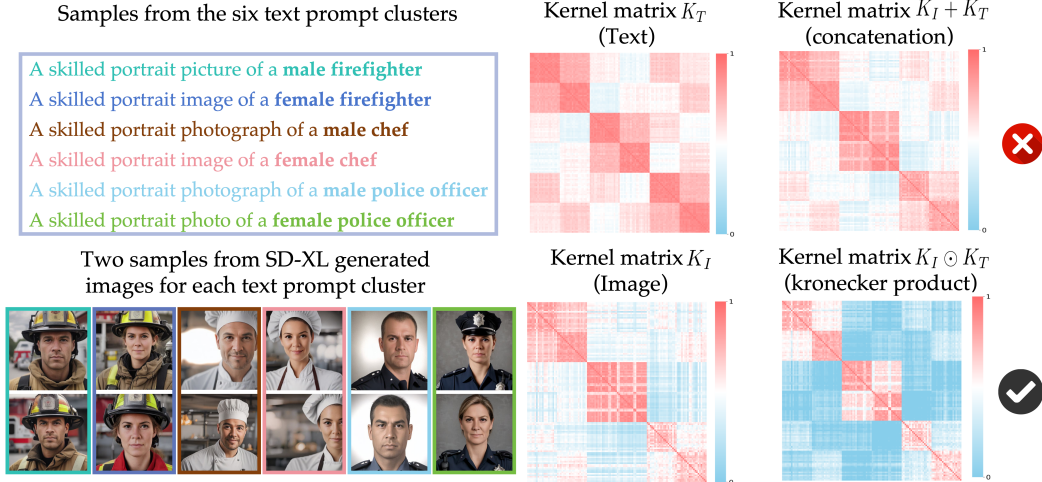


Figure 2: Kernel similarity heatmaps for (text,image) data with 6 underlying clusters. While the kernel matrix of the concatenated CLIP text and DINOv2 image embeddings blur cluster boundaries, the kernel matrices’ Hadamard product (for Kronecker-fused embedding) separates all the 6 groups.

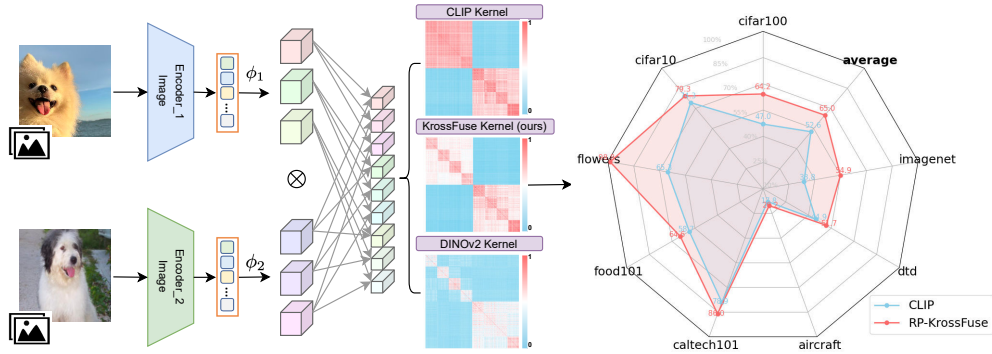


Figure 3: The Kronecker product fusion of embeddings in our proposed KrossFuse: The RP-KrossFuse fusion (implemented with Random Projection) of CLIP and DINOv2 could improve the averaged few-shot classification accuracy over CLIP on 9 benchmark image datasets.

We further apply the KrossFuse framework to address the performance gap between cross-modal and uni-modal embeddings. Cross-modal models such as CLIP, ALIGN [32], and BLIP [38] achieve alignment across modalities but often lag behind modality-specific experts such as DINOv2 and E5 [74] on domain-specific benchmarks. The embedding fusion in this setting poses a significant challenge: the uni-modal expert lacks an embedding map for the other modality (e.g., DINOv2 provides no text encoder). We demonstrate that KrossFuse naturally extends to this case by defining a *symmetrized embedding* for the shared modality and an *imputed constant map* for the missing modality, ensuring that the Kronecker product remains well-defined and balanced across domains.

We perform several experiments to demonstrate that RP-KrossFuse effectively fuses CLIP with uni-modal image (DINOv2) and text (Sentence-RoBERTa) embeddings, achieving competitive modality-specific accuracy while preserving strong cross-modal alignment. For example, Figure 3 shows the improved results of the KrossFuse fusion of DINOv2 and CLIP in few-shot classification over the CLIP model. Our results suggest the RP-KrossFuse framework provides a scalable, training-free mechanism to unify embedding structures. In summary, our contributions are:

- We propose *KrossFuse*, an embedding fusion framework that applies the principle of *kernel multiplication* to unify the cluster structures of embeddings.
- We develop *RP-KrossFuse*, a scalable, random-projection-based extension that efficiently approximates the Kronecker feature space.
- We apply KrossFuse to fuse cross-modal and uni-modal embeddings, demonstrating that it can enhance modality-specific performance while preserving cross-modal alignment.

2 Related Work

Cross-modal Embeddings. Recent advances in cross-modal embeddings have bridged the gap between visual and textual modalities. CLIP [60] is a pioneer model in this field using a contrastive learning approach, enabling remarkable zero-shot capabilities. Subsequent models, such as ALIGN [32] and FLORENCE [79], scaled datasets and refined learning objectives for improved performance. Variants like CoCa [78] introduced captioning objectives, while BLIP and BLIP-2 [38, 39] employed bootstrapping techniques to generate synthetic image-text pairs. Recent works, including OpenFlamingo [5], PaLI [10] and ImageBind [18], extended capabilities to few-shot scenarios and multilingual, multi-modal scaling. However, a trade-off could exist: strong cross-modal alignment often comes at the price of relatively lowering the performance in single modalities. Our work focuses on this trade-off and explores fusing cross-modal and uni-modal strengths into a unified embedding.

Combining Representation of Embeddings. Unifying the strengths of different embedding models has been studied in the representation learning literature. The EVA [15] and X-CLIP [45] methods enhance modality-specific performance while preserving cross-modal capabilities. Bertalign [44] aligns multilingual embeddings via parallel corpus supervision, and ImageBind [18] unifies six modalities using image-paired data. VLMo [6] balances modality-specific traits with cross-modal interaction, while FLAVA [67] and UniCLIP [37] propose unified frameworks for joint representation learning. Different from these training-based strategies, our method offers a training-free approach by applying the Kronecker product of the involved embeddings. By leveraging the symmetrized Kronecker product, our proposed KrossFuse aims to fuse cross-modal and uni-modal embeddings, preserving alignment while improving single-modality performance. To our knowledge, the task of fusing cross-modal and uni-modal embeddings has not been explored exclusively in the literature.

Random Projections and Kronecker Products. Random projection is a well-established method for dimensionality reduction. The Johnson-Lindenstrauss (JL) lemma [34] guarantees distance preservation in random projection with high probability. Sparse random projections [1] and fast JL transforms [2] have been shown to further improve the computational efficiency. The Kronecker product combined with random projection has not been utilized in embedding fusion in the context of cross-modal embeddings. In the existing literature, random projections are applied separately before computing the Kronecker product. We unify the two operations in the RP-KrossFuse approach by using the Hadamard product of random projected embeddings and demonstrate the approximation of the kernel matrix of the Kronecker product output.

Kernel Embedding Methods for Generative AI. Kernel embeddings are used for evaluation and guidance of generative models. The Kernel Inception Distance (KID) [8] introduced kernel-based evaluation, followed by entropy-based measures including diversity metrics [17, 55, 17, 31, 81, 54, 56, 82]. Kernel formulations are extended to distributed [75] and online evaluation settings [26, 25, 63], as well as explainable [29, 20, 19] and diversity-guided [30] embedding. It is relevant to explore how *KrossFuse* could be extended to these frameworks through its product-kernel formulation.

3 Preliminaries

3.1 Kernel Functions and Feature Maps

Consider a data vector $x \in \mathcal{X}$. A function $k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ is called a kernel function if for every $n \in \mathbb{N}$ and set of points $x_1, \dots, x_n \in \mathcal{X}$, the following kernel matrix $K_X \in \mathbb{R}^{n \times n}$ will be positive semi-definite (PSD):

$$K_X := \begin{bmatrix} k(x_1, x_1) & \cdots & k(x_1, x_n) \\ \vdots & \ddots & \vdots \\ k(x_n, x_1) & \cdots & k(x_n, x_n) \end{bmatrix}$$

The Moore-Aronszajn Theorem implies that k is a kernel function if there is a feature map $\phi : \mathcal{X} \rightarrow \mathbb{R}^s$ such that $k(x, y) = \langle \phi(x), \phi(y) \rangle$ is the inner product (denoted by $\langle \cdot, \cdot \rangle$) of the representations $\phi(x)$ and $\phi(y)$. An example is the linear kernel $k_{\text{lin}}(x, y) = x^\top y$ provided by the standard inner product. Another example is the Gaussian (RBF) kernel function with bandwidth parameter $B > 0$:

$$k(x, y) = \exp\left(-\frac{\|x - y\|_2^2}{B}\right)$$

The Schur product theorem shows that for every two kernel functions k_1, k_2 , their product $k(x, y) := k_1(x, y)k_2(x, y)$ will also be a kernel function. It can be seen that the feature map ϕ of the product kernel k is the Kronecker product of the feature maps ϕ_1, ϕ_2 of the kernels k_1, k_2 , i.e.:

$$\phi(x) = \phi_1(x) \otimes \phi_2(x)$$

In the above $\otimes$ denotes the Kronecker product which for matrix $A \in \mathbb{R}^{m \times d}$ with entries $a_{i,j}$'s and matrix $B \in \mathbb{R}^{s \times l}$ is defined as:

$$A \otimes B := \begin{bmatrix} a_{1,1}B & \cdots & a_{1,d}B \\ \vdots & \ddots & \vdots \\ a_{m,1}B & \cdots & a_{m,d}B \end{bmatrix} \in \mathbb{R}^{ms \times dl}$$

3.2 Cross-Modal Embedding Maps and Kernel Spaces

Consider two data domains $\mathcal{X}$ (e.g. for image modality) and $\mathcal{T}$ (e.g. for text modality). We call an embedding map $\psi = (\psi_X, \psi_T)$ cross-modal for these two domains if it offers modality-specific maps $\phi_X : \mathcal{X} \rightarrow \mathcal{Z}$ and $\phi_T : \mathcal{T} \rightarrow \mathcal{Z}$ that respectively map a vector $x \in \mathcal{X}$ in the first domain and a vector $t \in \mathcal{T}$ in the second domain to a shared embedding space $\mathcal{Z}$, i.e. we have $\psi_X(x), \psi_T(t) \in \mathcal{Z}$. The standard cross-modal embeddings are trained such that their outputs for relevant paired data point (x, t) is properly aligned. This property can be mathematically formulated as for a proper kernel function $k : \mathcal{Z} \times \mathcal{Z} \rightarrow \mathbb{R}$ the kernel function $k(\psi_X(x), \psi_T(t))$ is supposed to attain high values for relevant paired sample $(x, t) \sim P_{X,T}$ drawn from a ground-truth joint distribution $P_{X,T}$.

4 KrossFuse: Kronecker Fusion of Embeddings

4.1 Fusing Uni-modal Embeddings using their Kronecker Product

Consider two uni-modal embedding maps $\gamma_1 : \mathcal{X} \rightarrow \mathcal{Z}_1$ and $\gamma_2 : \mathcal{X} \rightarrow \mathcal{Z}_2$. To fuse γ_1 and γ_2 , we analyze the kernel similarity functions of the two embeddings. Considering kernel functions $k_1 : \mathcal{Z}_1 \times \mathcal{Z}_1 \rightarrow \mathbb{R}$ and $k_2 : \mathcal{Z}_2 \times \mathcal{Z}_2 \rightarrow \mathbb{R}$ operating in the embedding spaces, each of the embeddings γ_1 and γ_2 provide a kernel function for inputs $x, y \in \mathcal{X}$:

$$k_{\gamma_1}(x, y) = k_1(\gamma_1(x), \gamma_1(y)), \quad k_{\gamma_2}(x, y) = k_2(\gamma_2(x), \gamma_2(y)). \quad (2)$$

Note that if ϕ_1, ϕ_2 denote the feature maps of kernels k_1, k_2 , i.e. $k_1(x, y) = \langle \phi_1(x), \phi_1(y) \rangle$ and $k_2(x, y) = \langle \phi_2(x), \phi_2(y) \rangle$, then the feature map of k_{γ_1} and k_{γ_2} will be $\phi_1 \circ \gamma_1$ and $\phi_2 \circ \gamma_2$, respectively.

In fusing the embeddings, we set the fused kernel function to be the product of the marginal kernel functions $k_{\gamma_1} \cdot k_{\gamma_2}$. As shown in Figure 1, this implies that the similarity score between inputs x, y will be low if either of the kernel functions assign a low similarity score, i.e., that embedding distinguishes the input types. As a result, the inputs x, y will be clustered differently in the kernel-based method if their kernel similarity score is minor according to at least one of the embeddings. In the following proposition, we show that the Kronecker product of the embeddings' feature map $\phi_1 \circ \gamma_1$ and $\phi_2 \circ \gamma_2$ possesses the mentioned kernel product property. We defer the proof of the theoretical statements to the Appendix.

Proposition 1. *Consider feature maps $\phi_1 : \mathcal{Z}_1 \rightarrow \mathbb{R}^{d_1}$, $\phi_2 : \mathcal{Z}_2 \rightarrow \mathbb{R}^{d_2}$ and their corresponding kernel functions k_1, k_2 . Then, given the kernel functions defined in (2), the product kernel function $k_{\gamma_1}(x, y) \cdot k_{\gamma_2}(x, y)$ has the feature map $\phi_{\gamma_1, \gamma_2} : \mathcal{X} \rightarrow \mathbb{R}^{d_1 d_2}$ defined using the Kronecker product:*

$$\phi_{\gamma_1, \gamma_2}(x) = \phi_1(\gamma_1(x)) \otimes \phi_2(\gamma_2(x))$$

Therefore, the feature map to combine the two embeddings follows from the Kronecker multiplication of $\phi_1 \circ \gamma_1$ and $\phi_2 \circ \gamma_2$, which maps an input $x \in \mathcal{X}$ to a space of dimension $d_1 d_2$, i.e. the product of the dimensions of maps ϕ_1 and ϕ_2 .

Remark 1. *The discussed Kronecker product combination of two embeddings can be further extended to multiple m embeddings $\gamma_1, \dots, \gamma_m$. Assuming kernel functions $k_1, k_2, \dots, k_m$ (with feature maps*

$\phi_1, \dots, \phi_m$) to operate on the m embedded data, the feature map corresponding to the unified product kernel $\prod_{i=1}^m k_i(\gamma_i(x), \gamma_i(y))$ will be

$$\phi_{\gamma_1, \dots, \gamma_m}(x) = \bigotimes_{i=1}^k \phi_i(\gamma_i(x)) \in \mathbb{R}^{d_1 d_2 \dots d_m}$$

The above implies that, using the above feature map, the resulting kernel function could distinguish the dissimilarity of inputs x, y , if one of the embeddings can differentiate the two data points.

4.2 Extending KrossFuse for Kronecker Fusion of Uni-modal and Cross-Modal Embeddings

We earlier discussed how to fuse unimodal representations γ_1 and γ_2 via their Kronecker product, such that the kernel similarity function of the fused embedding is the product of their kernels. However, the challenge in combining a cross-modal embedding $\psi = (\psi_X, \psi_T)$ operating on two modalities in $\mathcal{X}, \mathcal{T}$ and a uni-modal embedding $\gamma = (\gamma_X)$ of only the modality $\mathcal{X}$ is the missing operator of γ to apply to the non-shared modality $\mathcal{T}$. For example, if we suppose ψ represents the CLIP cross-modal model applying to image $\mathcal{X}$ and text $\mathcal{T}$ domains and γ denotes the DINOv2 embedding applying to single image $\mathcal{X}$, then we do not have the text part γ_T to multiply to the text part of CLIP model.

To apply the Kronecker product-based fusion of the embeddings, we propose the *KrossFuse* method, where we define the following symmetrized cross-modal embedding $\tilde{\gamma} = (\tilde{\phi}_{\gamma, X}, \tilde{\phi}_{\gamma, T})$ to play the role of the uni-modal embedding $\gamma = (\gamma_X)$ in the Kronecker fusion process:

$$\begin{aligned} \tilde{\phi}_{\gamma, X}(x) &:= \frac{1}{\sqrt{2}} \left[\sqrt{\frac{C}{d}} + \phi(\gamma_X(x)), \sqrt{\frac{C}{d}} - \phi(\gamma_X(x)) \right]^\top \\ \tilde{\phi}_{\gamma, T}(t) &:= \sqrt{\frac{C}{2d}} \cdot \underbrace{[1, \dots, 1]}_{2d \text{ times}}^\top \end{aligned} \quad (3)$$

Here, $C > 0$ is defined as a hyperparameter constant determining the constant similarity score of every two data points in the missing modality. Note that ϕ denotes the feature map of the given kernel function for the single-modality embedding, and d denotes the dimension of ϕ 's output vector.

Given the symmetrized cross-modal embedding $\tilde{\gamma}$ which also applies to the non-shared modality $\mathcal{T}$, KrossFuse combines the cross-modal embedding $\psi = (\psi_X, \psi_T)$ (e.g. CLIP) and the uni-modal embedding γ (e.g. DINOv2) by taking the Kronecker product of ψ and $\tilde{\gamma}$ in each modality as:

$$E_X(x) := \phi(\Psi_X(x)) \otimes \tilde{\phi}_{\gamma, X}(x), \quad (4)$$

$$E_T(t) := \phi(\Psi_T(t)) \otimes \tilde{\phi}_{\gamma, T}(t) \quad (5)$$

Proposition 2. *Given the combined cross-modal embedding in (4) and kernel function $k(x, y) = \langle \phi(x), \phi(y) \rangle$ for feature map ϕ , the following inner product will hold for every inputs $x, x' \in \mathcal{X}$ from the shared modality and inputs $t, t' \in \mathcal{T}$ from the non-shared modality:*

$$\begin{aligned} \langle E_X(x), E_X(x') \rangle &= k(\psi_X(x), \psi_X(x')) \left(C + k(\gamma_X(x), \gamma_X(x')) \right), \\ \langle E_T(t), E_T(t') \rangle &= C \cdot k(\psi_T(t), \psi_T(t')), \\ \langle E_X(x), E_T(t) \rangle &= C \cdot k(\psi_X(x), \psi_T(t)). \end{aligned}$$

As a result, for the merged KrossFuse embedding, the inner product (i.e. the resulting kernel function) between the transformation of two inputs from either of the modalities will be the product of C and the kernel function of the cross-modal embedding (e.g. CLIP model), except the case of two inputs from the shared embedding where the value will be added to the product of the kernels from both the cross-modal and uni-modal embeddings.

Remark 2. *The KrossFuse framework can be similarly applied to fuse a cross-modal embedding with multiple uni-modal embeddings. Such a multi-embedding fusion involves the Kronecker product of the modified embedding for every uni-modal embedding in the unification. Therefore, the KrossFuse algorithm can be applied to fuse each of the modalities of a cross-modal embedding with specialized uni-modal models. For example, each of the text and image embedding of CLIP can be merged with text-specific (e.g. E5) and image-specific (e.g. DINOv2) models.*

Algorithm 1 Kernel Feature Fusion of Two Embeddings

```
1: Input: Image samples  $\{x_i\}_{i=1}^{N_x}$ , First image encoder  $\gamma_1$ , Second image encoder  $\gamma_2$ , Kernel maps  $\phi_1, \phi_2$ , Projected dim  $l$ 
2:  $U_1 \sim \text{Uniform}[-\sqrt{3}, \sqrt{3}]^{d_{\phi_1} \times l} / \sqrt{l}$ 
3:  $U_2 \sim \text{Uniform}[-\sqrt{3}, \sqrt{3}]^{d_{\phi_2} \times l} / \sqrt{l}$ 
4: Initialize  $Z^{\text{img}} \in \mathbb{R}^{N_x \times l}$ 
5: for batch  $\mathcal{B}_x$  in  $\{x_i\}$  do
6:    $\psi_{1,\mathcal{B}_x} \leftarrow \phi_1(\gamma_1(\mathcal{B}_x)), \psi_{2,\mathcal{B}_x} \leftarrow \phi_2(\gamma_2(\mathcal{B}_x))$ 
7:    $Z_{\mathcal{B}_x}^{\text{img}} \leftarrow (\psi_{1,\mathcal{B}_x} U_1) \odot (\psi_{2,\mathcal{B}_x} U_2)$ 
8: end for
9: Return  $Z^{\text{img}}$ 
```

5 RP-KrossFuse: Scalable Embedding Fusion via Random Projection

Although the Kronecker product in KrossFuse can combine multiple embeddings, the dimension of the merged feature vector will be the product of the dimension of individual models, which would be computationally challenging in standard applications. To lower the computational costs, we propose a scalable application of random projection that can preserve the inner product (kernel function values) with limited feature size. The proposed extension, which we call RP-KrossFuse, applies random projection to each cross-modality embedding (modified cross-modality embedding for an original uni-modal embedding). To do this, for each of the cross-modality embeddings $\psi_1 = (\psi_{X,1}, \psi_{T,1})$ and $\psi_2 = (\psi_{X,2}, \psi_{T,2})$, we generate random matrix a $U_i \in \mathbb{R}^{l \times d_i}$ whose entries are independent random variables with uniform distribution over $[-\sqrt{3}, \sqrt{3}]$, that has unit variance. Then, the RP-KrossFuse embedding of each of inputs $x \in \mathcal{X}$ and $t \in \mathcal{T}$ will be

$$\tilde{\psi}_X(x) = \frac{1}{\sqrt{l}}(U_1 \psi_{1,X}(x)) \odot (U_2 \psi_{2,X}(x)) \quad (6)$$

$$\tilde{\psi}_T(t) = \frac{1}{\sqrt{l}}(U_1 \psi_{1,T}(t)) \odot (U_2 \psi_{2,T}(t)) \quad (7)$$

In the above, $\odot$ denotes the element-wise Hadamard product. This formulation leads to Algorithm 1 for uni-modal fusion and Algorithm 2 for cross-modal fusion. Note that computing the above RP-KrossFuse embeddings requires $\mathcal{O}(l(d_1 + d_2))$ operations. Also, the output has l dimensions, which for a properly bounded l will be significantly cheaper than $d_1 d_2$ dimensions in the Kronecker product of the two embedded vectors. Theorem 1 shows the above method can preserve the KrossFuse inner products with high probability.

Theorem 1. Consider n input pairs $(t_i, x_i)_{i=1}^n$. Suppose $\max\{\|\psi_{X,j}(x_i)\|^2, \|\psi_{T,j}(t_i)\|^2\} \leq B$ is norm-bounded for every $j \in \{1, 2\}$ and index $i \in \{1, \dots, n\}$. Then, for any $\delta > 0$, we have with probability at least $1 - \delta$:

$$\begin{aligned} \left| \tilde{\psi}_T(t_i)^\top \tilde{\psi}_T(t_j) - \psi_{1,2,T}(t_i)^\top \psi_{1,2,T}(t_j) \right| &\leq \sqrt{\frac{2B^2 \log(2n^2/\delta)}{l}}, \\ \left| \tilde{\psi}_X(x_i)^\top \tilde{\psi}_X(x_j) - \psi_{1,2,X}(x_i)^\top \psi_{1,2,X}(x_j) \right| &\leq \sqrt{\frac{2B^2 \log(2n^2/\delta)}{l}} \end{aligned}$$

In the Appendix, we have also proved Theorem 2 on the extension of the random projection approach to infinite-dimensional shift-invariant kernels, e.g., RBF kernels, via the framework of random Fourier features [61]. We defer the discussion of the shift-invariant kernels to the Appendix A.4.

6 Numerical Results

To evaluate the effectiveness of our proposed fusion methods, KrossFuse and RP-KrossFuse, we performed several numerical experiments regarding unimodal and cross-modal embedding tasks (see appendix B for implementation details). We tested the proposed and baseline fusion methods in application to the following embedding models: cross-modal (image,text) embeddings of CLIP [60], OpenCLIP [28] and SigLIP [80], image modality embeddings of DINOv2 [53], Unicom [3], and

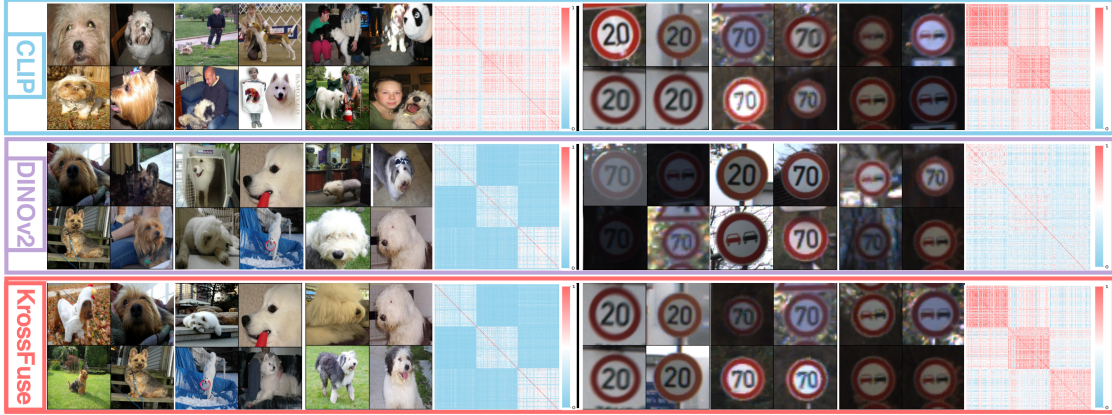


Figure 4: Clustering results and kernel matrix heatmaps for CLIP, DINOv2, and KrossFuse on ImageNet dog breeds and GTSRB dataset. While CLIP could not fully separate all the dog categories and DINOv2 struggled in clustering traffic signs, the KrossFuse fusion captured the clusters.

text modality embeddings of S-RoBERTa [62], E5 [74]. Unless otherwise specified, the additional numerical results are provided in appendix C.

Visualization of kernel similarity matrix for the Kronecker product of embeddings. To visualize how the Kronecker product of two embeddings can capture the clusters identified by each of the embeddings, we present the numerical results of performing kernel-based clustering on the following image datasets: CUB-200-2011 [73], Oxford Flowers [51], DTD [11], Image-Woof [23] consisting of ten dog breeds from ImageNet-1K [13], GTSRB [69] and typographic attack images by introducing mislabeled red text into 10 ImageNet subclasses following the reference [47]. The visualizations of the kernel matrices for these datasets highlight how CLIP and DINOv2 could assign differently structured kernel similarity scores in different image types. We present three representative image clustering results, visualized with kernel matrix heatmaps in Figure 4. In the Image-Woof dataset, the DINOv2 similarity map could visually capture the existing three clusters, while the CLIP embedding did not completely separate the three clusters. In contrast, on the three traffic sign classes from GTSRB dataset, the CLIP embedding could differentiate the three groundtruth clusters in its kernel matrix, while the DINOv2 assigned kernel matrix did not display the three clusters. On the other hand, the Kronecker product kernel matrix in KrossFuse shows clear block-diagonal structures on both datasets.

Unimodal Classification Results. To assess the performance of the RP-Krossfuse fusion of representations CLIP and DINOv2, we performed linear probe classification on the following standard image datasets: ImageNet [13], GTSRB [69], and SVHN [50], as well as on out-of-distribution (OOD) benchmarks: ImageNet-A [22] and ImageNet-R [21]. The results are shown in Table 1, where KrossFuse is compared with the four baselines (see detailed implementation in appendix B.2: (1) KPoMRP, which utilizes the Kronecker product of marginal random projections; (2) GATE, a simplified implementation of the Mixture-of-Experts (MoE) [66] paradigm; (3) ATTN, a lightweight feature-level attentional fusion method [84]; and (4) COMM, a MLP projector fusion framework proposed in [33]. Notably, RP-KrossFuse obtained 84.1% accuracy on ImageNet, which was improving upon the DINOv2, CLIP image embeddings and also the baselines. While DINOv2 performed competitively on OOD benchmarks, RP-KrossFuse reached better accuracy scores on GTSRB and SVHN. We note that CLIP and DINOv2 would likely excel on certain image categories, and the RP-KrossFuse method seems to consistently fuse the strengths of the two embeddings.

For the text experiments, we used the SentEval toolkit [12] to evaluate the RP-KrossFuse embeddings compared to CLIP and S-RoBERTa on the following NLP classification benchmarks: MR [58], CR [24], SUBJ [57], MPQA [76], SST2 [68], TREC [72], and MRPC [14]. Comprehensive evaluation can be found in table 2.

Zero-shot Cross-modal Alignment. To test whether the improved unimodal performance of our fused embeddings comes without compromising CLIP’s zero-shot cross-modal alignment, we visualized the cosine similarity distributions of positive and negative image-text pairs on the MSCOCO

Table 1: Linear probe evaluation of embeddings on various image benchmarks (IN: ImageNet).

[†]Projection dimension of RP-KrossFuse and KPoMRP: typically 3000 (except 5000 for IN).

Embedding	IN	GTSRB	SVHN	IN-A	IN-R
CLIP [60]	73.2	83.1	63.6	23.2	60.0
DINOv2 [53]	83.3	72.5	60.5	48.5	68.8
GATE [66]	81.4	82.2	66.0	38.9	59.1
ATTN [84]	79.5	77.3	64.6	38.9	61.5
KPoMRP [†]	79.4	71.8	49.4	34.8	55.2
COMM [33]	82.7	76.7	65.5	44.7	63.3
RP-KrossFuse [†]	84.1	82.7	66.9	47.6	67.4

Table 2: Linear probe evaluation of frozen features of variants of CLIP, Sroberta, RP-KrossFuse and three baselines using the SentEval toolkit on text benchmarks. Test accuracy (%) are based on a 5-fold cross-validation. The projection dimension of KPoMRP and RP-KrossFuse is 3000.

Embedding	Arch	Fused	MR	CR	SUBJ	MPQA	SST2	TREC	MRPC	Avg
CLIP [60]	ViT-B/32	✗	75.8	83.1	92.5	86.4	82.0	83.0	70.1	81.8
	ViT-L/14	✗	78.1	85.3	93.8	87.0	83.9	86.4	67.7	83.2
KPoMRP	ViT-B/32	✓	73.8	78.5	86.6	82.1	80.3	74.8	70.8	78.1
	ViT-L/14	✓	72.5	80.3	86.0	83.1	74.0	78.2	68.1	77.5
GATE [66]	ViT-B/32	✓	84.8	87.2	94.4	88.5	91.8	89.3	65.5	85.9
	ViT-L/14	✓	84.3	87.8	94.5	88.3	91.4	89.3	67.6	86.2
ATTN [84]	ViT-B/32	✓	85.7	86.3	93.4	88.8	91.9	88.5	66.2	85.8
	ViT-L/14	✓	85.7	85.9	94.3	87.8	92.4	86.5	65.9	85.5
RP-KrossFuse	ViT-B/32	✓	85.8	88.7	94.4	89.1	89.7	95.0	73.6	88.0
	ViT-L/14	✓	86.0	88.1	94.8	89.3	89.8	95.2	73.6	88.1
SRoBERTa [62]	TF-L24	✗	85.1	86.8	93.7	87.7	89.1	93.2	68.1	86.2

validation dataset in Figure 5. The overlapping curves suggest that KrossFuse could preserve the geometric distribution of both positive and negative image text pairs. We report zero-shot image-to-text and text-to-image retrieval results on MSCOCO [42] and Flickr30k [77] in table 6 of the appendix. We can observe that the differences between CLIP and RP-KrossFuse are mostly below 1%, suggesting that RP-KrossFuse maintains strong zero shot cross-modal alignment, with retrieval performance comparable to CLIP.

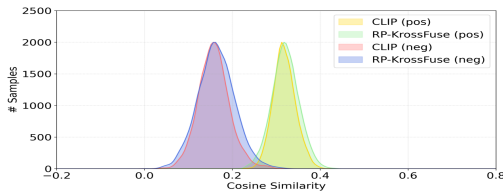


Figure 5: The cosine similarity distributions in MSCOCO.

for the text modality. To further show the benefits of enhanced text representations, we extend this evaluation to the MSCOCO dataset, where ground-truth captions can be directly used as text samples. As shown in Table 3, our method of cross modality could perform better than CLIP and the other baselines, particularly in 1 and 2 shot cases, highlighting the gains reached by the fusion method in cross-modal transfer learning.

Ablation Studies. To test the effect of the different RP-KrossFuse components, we evaluated the classification accuracy on ImageNet and the average accuracy across seven NLP benchmarks in SentEval. As shown in Figure 6: (a) Fusing CLIP with image expert embeddings (UniCom, DINOv2) could also improve performance over the individual CLIP embedding. (b) Fusing text-

Cross-modal Few-shot Learning. To evaluate how RP-KrossFuse applies to cross-modal few-shot learning scenarios, we used samples from one modality to enhance few shot learning in another modality following the work in [43] which simulates data-limited learning tasks. This cross-modal adaptation task is enabled by multimodal foundation models like CLIP, which map different modalities into a shared representation space, thereby allowing text samples to augment image samples. For generic object and scene image benchmarks, we adopt the standard prompt "a photo of a [class]"

Table 3: Cross-modal few-shot classification results across datasets. "Ours" = RP-KrossFuse (proj. dim. 3000); "I"/"T" denote image/text domains.

Shots	Method	Caltech [16]	Food [9]	DTD [11]	Aircraft [46]	ImageNet [13]	MSCOCO [42]	Average
1	CLIP [60] (I)	70.9	37.8	35.4	14.6	24.3	8.7	32.0
	CLIP [60] (I+T)	78.9	58.7	44.9	17.8	33.8	31.6	44.3
	DINOv2 [53] (I)	84.3	57.9	47.2	15.4	54.0	16.4	45.8
	Ours (I)	84.6	55.7	48.3	19.4	51.8	21.5	46.9
	Ours (I+T)	86.0	64.6	51.7	20.3	54.9	43.5	53.5
2	CLIP [60] (I)	78.9	47.8	44.2	18.2	30.2	11.2	38.4
	CLIP [60] (I+T)	82.7	60.7	47.3	19.8	36.0	47.2	49.0
	DINOv2 [53] (I)	88.3	63.4	57.3	17.3	61.9	23.1	51.9
	Ours (I)	89.2	63.6	57.3	23.6	60.9	36.8	55.2
	Ours (I+T)	90.1	68.0	59.5	24.8	62.1	51.5	59.3
4	CLIP [60] (I)	83.3	57.7	51.9	20.6	36.8	23.9	45.7
	CLIP [60] (I+T)	84.6	64.8	52.0	21.1	42.4	57.5	53.7
	DINOv2 [53] (I)	90.4	69.8	64.0	20.9	67.0	38.5	58.4
	Ours (I)	90.8	71.8	64.4	28.1	66.6	52.8	62.4
	Ours (I+T)	91.1	73.8	65.0	28.2	67.2	58.1	63.9
8	CLIP [60] (I)	84.5	65.5	53.7	24.2	42.1	44.9	52.5
	CLIP [60] (I+T)	85.8	68.7	54.6	24.6	45.2	61.2	56.7
	DINOv2 [53] (I)	91.4	73.0	69.2	24.5	70.4	53.6	63.7
	Ours (I)	92.0	75.9	69.2	31.7	70.4	55.1	65.7
	Ours (I+T)	92.2	76.9	69.4	31.7	70.7	61.9	67.1

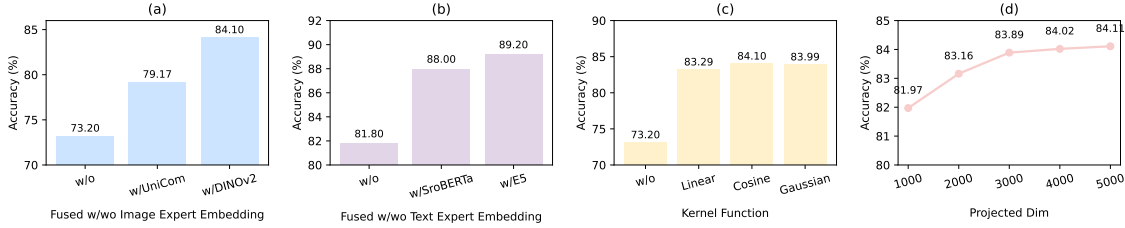


Figure 6: Ablation Studies. (a) (b) Effect of fusing different image and text expert embeddings. (c) Effect of kernel function. (d) Effect of random projected dimension.

expert embeddings (S-RoBERTa, E5) could bring considerable gains compared to the CLIP text encoder. (c) All kernel-based fusions led to performance improvements, with the Cosine and RBF kernels yielding slightly higher gains than the linear kernel. (d) Increasing the random projection dimension would lead to gradually saturated accuracy, suggesting RP-KrossFuse could converge to KrossFuse as the dimension reaches around 3000.

7 Conclusion and Limitations

The proliferation of powerful embedding models across vision, language, and other modalities underscores the need for principled methods to unify complementary representations. This work introduced KrossFuse, a Kronecker-product framework for embedding fusion grounded in the kernel product principle, and its scalable variant RP-KrossFuse, which employs random projection to efficiently approximate the high-dimensional Kronecker feature space. The proposed formulation provides a simple, training-free mechanism to integrate embeddings from diverse sources—such as cross-modal and domain-specific models—while preserving the discriminative structure of each.

Despite its efficiency and generality, RP-KrossFuse introduces a few additional hyperparameters, primarily the scaling coefficient C and the projection dimension l . The fused embedding may require a higher projection dimension (e.g., 3,000) than individual encoders such as CLIP or DINOv2. While this difference may affect direct dimensional comparability, it aligns with standard practice in representation learning, where embeddings of varying sizes are routinely compared. The use of random projection is a deliberate design choice that enables scalability without the prohibitive cost of operating in the full Kronecker feature space (on the order of 8×10^5 dimensions). When strict dimensional parity or compactness is desired, post-hoc dimensionality reduction via PCA can be applied to the fused embedding. Future work may extend this framework to non-visual modalities, multi-way fusion scenarios, or attention-based adaptation modules that refine the fused embedding for task-specific objectives.

Acknowledgments

The work of Farzan Farnia is partially supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China, Project 14209920, and is partially supported by CUHK Direct Research Grants with CUHK Project No. 4055164 and 4937054. The authors acknowledge the support from the Hong Kong Research Grants Council (RGC) and the Hong Kong PhD Fellowship Scheme (HKPFS) award supporting Youqi Wu’s research. Also, the authors sincerely thank the anonymous reviewers for their insightful comments and constructive suggestions.

References

- [1] Dimitris Achlioptas. Database-friendly random projections: Johnson-lindenstrauss with binary coins. *Journal of computer and System Sciences*, 66(4):671–687, 2003.
- [2] Nir Ailon and Bernard Chazelle. The fast johnson–lindenstrauss transform and approximate nearest neighbors. *SIAM Journal on computing*, 39(1):302–322, 2009.
- [3] Xiang An, Jiankang Deng, Kaicheng Yang, Jaiwei Li, Ziyong Feng, Jia Guo, Jing Yang, and Tongliang Liu. Unicom: Universal and compact representation learning for image retrieval. *arXiv preprint arXiv:2304.05884*, 2023.
- [4] Reyhane Askari Hemmat, Melissa Hall, Alicia Sun, Candace Ross, Michal Drozdal, and Adriana Romero-Soriano. Improving geo-diversity of generated images with contextualized vendi score guidance. In *European Conference on Computer Vision*, pages 213–229. Springer, 2024.
- [5] Anas Awadalla, Irena Gao, Josh Gardner, Jack Hessel, Yusuf Hanafy, Wanrong Zhu, Kalyani Marathe, Yonatan Bitton, Samir Gadre, Shiori Sagawa, et al. Openflamingo: An open-source framework for training large autoregressive vision-language models. *arXiv preprint arXiv:2308.01390*, 2023.
- [6] Hangbo Bao, Wenhui Wang, Li Dong, Qiang Liu, Owais Khan Mohammed, Kriti Aggarwal, Subhojit Som, Songhao Piao, and Furu Wei. Vlmo: Unified vision-language pre-training with mixture-of-modality-experts. *Advances in Neural Information Processing Systems*, 35: 32897–32912, 2022.
- [7] Alain Berlinet and Christine Thomas-Agnan. *Reproducing Kernel Hilbert Spaces in Probability and Statistics*. Springer, New York, 2004. ISBN 978-1-4419-9096-9.
- [8] Mikołaj Bińkowski, Danica J. Sutherland, Michael Arbel, and Arthur Gretton. Demystifying MMD GANs. In *International Conference on Learning Representations*, 2018. doi: 10.48550/arXiv.1801.01401.
- [9] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101—mining discriminative components with random forests. In *European Conference on Computer Vision*, pages 446–461. Springer, 2014.
- [10] Xi Chen, Xiao Wang, Soravit Changpinyo, AJ Piergiovanni, Piotr Padlewski, Daniel Salz, Sebastian Goodman, Adam Grycner, Basil Mustafa, Lucas Beyer, et al. Pali: A jointly-scaled multilingual language-image model. *arXiv preprint arXiv:2209.06794*, 2022.
- [11] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3606–3613, 2014.
- [12] Alexis Conneau and Douwe Kiela. Senteval: An evaluation toolkit for universal sentence representations. *arXiv preprint arXiv:1803.05449*, 2018.
- [13] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

- [14] William B Dolan and Chris Brockett. Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the Third International Workshop on Paraphrasing (IWP2005)*, 2005.
- [15] Yuxin Fang, Wen Wang, Binhui Xie, Quan Sun, Ledell Wu, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. Eva: Exploring the limits of masked visual representation learning at scale. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19358–19369, 2023.
- [16] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *2004 Conference on Computer Vision and Pattern Recognition Workshop*, pages 178–178. IEEE, 2004.
- [17] Dan Friedman and Adji Bousso Dieng. The vendi score: A diversity evaluation metric for machine learning. *arXiv preprint arXiv:2210.02410*, 2022.
- [18] Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. Imagebind: One embedding space to bind them all. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15180–15190, 2023.
- [19] Shizhan Gong, Yankai Jiang, Qi Dou, and Farzan Farnia. Kernel-based unsupervised embedding alignment for enhanced visual representation in vision-language models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 19912–19931. PMLR, 13–19 Jul 2025.
- [20] Shizhan Gong, Haoyu Lei, Qi Dou, and Farzan Farnia. Boosting the visual interpretability of clip via adversarial fine-tuning. In *International Conference on Learning Representations*, 2025. URL <https://openreview.net/forum?id=khuIvzxPRp>.
- [21] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8340–8349, 2021.
- [22] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15262–15271, 2021.
- [23] Jeremy Howard. Imagewoof: a subset of 10 classes from imagenet that aren’t so easy to classify. <https://github.com/fastai/imagenette#imagewoof>, March 2019. GitHub repository.
- [24] Mingqing Hu and Bing Liu. Mining and summarizing customer reviews. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 168–177. ACM, 2004.
- [25] Xiaoyan Hu, Ho-Fung Leung, and Farzan Farnia. A multi-armed bandit approach to online selection and evaluation of generative models. In *Proceedings of The 28th International Conference on Artificial Intelligence and Statistics*, volume 258 of *Proceedings of Machine Learning Research*, pages 1864–1872. PMLR, 2025.
- [26] Xiaoyan Hu, Ho-Fung Leung, and Farzan Farnia. PAK-UCB contextual bandit: An online learning approach to prompt-aware selection of generative models and LLMs. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 24447–24481. PMLR, 2025.
- [27] Lawrence Hubert and Phipps Arabie. Comparing partitions. *Journal of classification*, 2:193–218, 1985.
- [28] Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. Openclip, July 2021.

- [29] Mohammad Jalali, Bahar Dibaei Nia, and Farzan Farnia. Towards an explainable comparison and alignment of feature embeddings. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 26757–26796. PMLR, 2025.
- [30] Mohammad Jalali, Haoyu Lei, Amin Gohari, and Farzan Farnia. Sparke: Scalable prompt-aware diversity and novelty guidance in diffusion models via RKE score. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.
- [31] Mohammad Jalali, Azim Ospanov, Amin Gohari, and Farzan Farnia. An information-theoretic approach to diversity evaluation of prompt-based generative models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pages 7634–7643, October 2025.
- [32] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. *International Conference on Machine Learning*, pages 4904–4916, 2021.
- [33] Dongsheng Jiang, Yuchen Liu, Songlin Liu, Jin’e Zhao, Hao Zhang, Zhen Gao, Xiaopeng Zhang, Jin Li, and Hongkai Xiong. From clip to dino: Visual encoders shout in multi-modal large language models, 2024.
- [34] William B Johnson, Joram Lindenstrauss, and Gideon Schechtman. Extensions of lipschitz maps into banach spaces. *Israel Journal of Mathematics*, 54(2):129–138, 1986.
- [35] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.
- [36] Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models. *Advances in neural information processing systems*, 32, 2019.
- [37] Janghyeon Lee, Jongsuk Kim, Hyounguk Shon, Bumsu Kim, Seung Hwan Kim, Honglak Lee, and Junmo Kim. Unclip: Unified framework for contrastive language-image pre-training. *Advances in Neural Information Processing Systems*, 35:1008–1019, 2022.
- [38] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*, pages 12888–12900. PMLR, 2022.
- [39] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models, 2023. URL <https://arxiv.org/abs/2301.12597>.
- [40] Xianhang Li, Yanqing Liu, Haoqin Tu, Hongru Zhu, and Cihang Xie. Openvision: A fully-open, cost-effective family of advanced vision encoders for multimodal learning. *arXiv preprint arXiv:2505.04601*, 2025.
- [41] Victor Weixin Liang, Yuhui Zhang, Yongchan Kwon, Serena Yeung, and James Y Zou. Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning. *Advances in Neural Information Processing Systems*, 35:17612–17625, 2022.
- [42] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, pages 740–755. Springer, 2014.
- [43] Zhiqiu Lin, Samuel Yu, Zhiyi Kuang, Deepak Pathak, and Deva Ramanan. Multimodality helps unimodality: Cross-modal few-shot learning with multimodal models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19325–19337, 2023.

- [44] Lei Liu and Min Zhu. Bertalign: Improved word embedding-based sentence alignment for chinese–english parallel corpora of literary texts. *Digital Scholarship in the Humanities*, 38(2): 621–634, 12 2022. ISSN 2055-7671. doi: 10.1093/llc/fqac089.
- [45] Yiwei Ma, Guohai Xu, Xiaoshuai Sun, Ming Yan, Ji Zhang, and Rongrong Ji. X-clip: End-to-end multi-grained contrastive learning for video-text retrieval. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 638–647, 2022.
- [46] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013.
- [47] Joanna Materzyńska, Antonio Torralba, and David Bau. Disentangling visual and written concepts in clip. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16410–16419, 2022.
- [48] Aaron F. McDaid, Derek Greene, and Neil Hurley. Normalized mutual information to evaluate overlapping community finding algorithms, 2013.
- [49] Muhammad Ferjad Naeem, Seong Joon Oh, Youngjung Uh, Yunjey Choi, and Jaejun Yoo. Reliable fidelity and diversity metrics for generative models. In *International conference on machine learning*, pages 7176–7185. PMLR, 2020.
- [50] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Baolin Wu, Andrew Y Ng, et al. Reading digits in natural images with unsupervised feature learning. In *NIPS workshop on deep learning and unsupervised feature learning*, volume 2011, page 4. Granada, 2011.
- [51] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *2008 Sixth Indian conference on computer vision, graphics & image processing*, pages 722–729. IEEE, 2008.
- [52] Teng Niu, Shiai Zhu, Lei Pang, and Abdulmotaleb El Saddik. Sentiment analysis on multi-view social data. In *MultiMedia Modeling: 22nd International Conference, MMM 2016, Miami, FL, USA, January 4-6, 2016, Proceedings, Part II 22*, pages 15–27. Springer, 2016.
- [53] Maxime Oquab, Timothée Darcet, Theo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.
- [54] Azim Ospanov and Farzan Farnia. Do vendi scores converge with finite samples? truncated vendi score for finite-sample convergence guarantees. In *Proceedings of the Forty-first Conference on Uncertainty in Artificial Intelligence*, volume 286 of *Proceedings of Machine Learning Research*, pages 3272–3299. PMLR, 2025.
- [55] Azim Ospanov, Jingwei Zhang, Mohammad Jalali, Xuenan Cao, Andrej Bogdanov, and Farzan Farnia. Towards a scalable reference-free evaluation of generative models. In *Advances in Neural Information Processing Systems*, 2024.
- [56] Azim Ospanov, Mohammad Jalali, and Farzan Farnia. Scendi score: Prompt-aware diversity evaluation via schur complement of clip embeddings. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 16927–16937, October 2025.
- [57] Bo Pang and Lillian Lee. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics*, pages 271–278, 2004.
- [58] Bo Pang and Lillian Lee. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, pages 115–124, 2005.
- [59] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3498–3505. IEEE, 2012.

- [60] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. *International conference on machine learning*, pages 8748–8763, 2021.
- [61] Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. *Advances in neural information processing systems*, 20, 2007.
- [62] N Reimers. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*, 2019.
- [63] Parham Rezaei, Farzan Farnia, and Cheuk Ting Li. Be more diverse than the most diverse: Optimal mixtures of generative models via mixture-UCB bandit algorithms. In *The Thirteenth International Conference on Learning Representations*, 2025.
- [64] Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.
- [65] Bernhard Schölkopf and Alexander J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Cambridge, MA, 2002. ISBN 978-0-262-19475-4.
- [66] Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarczyk, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538*, 2017.
- [67] Amanpreet Singh, Ronghang Hu, Vedanuj Goswami, Guillaume Couairon, Wojciech Galuba, Marcus Rohrbach, and Douwe Kiela. Flava: A foundational language and vision alignment model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15638–15650, 2022.
- [68] Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, 2013.
- [69] Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and Christian Igel. Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. *Neural networks*, 32: 323–332, 2012.
- [70] Pavan Kumar Anasosalu Vasu, Hadi Pouransari, Fartash Faghri, Raviteja Vemulapalli, and Oncel Tuzel. Mobileclip: Fast image-text models through multi-modal reinforced training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15963–15974, 2024.
- [71] Nguyen Xuan Vinh, Julien Epps, and James Bailey. Information theoretic measures for clusterings comparison: is a correction for chance necessary? In *Proceedings of the 26th annual international conference on machine learning*, pages 1073–1080, 2009.
- [72] Ellen M Voorhees and Donna M Tice. Building a question answering test collection. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 200–207. ACM, 2000.
- [73] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011.
- [74] Liang Wang, Nan Yang, Furu Wei, Xiaojun Huang, and Binxing Ma. Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv:2212.03533*, 2023.
- [75] Zixiao Wang, Farnia Farzan, Zhenghao Lin, Yunheng Shen, and Bei Yu. On the distributed evaluation of generative models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 7644–7653, 2025.

- [76] Janyce Wiebe, Theresa Wilson, and Claire Cardie. Annotating expressions of opinions and emotions in language. *Language Resources and Evaluation*, 39(2-3):165–210, 2005.
- [77] Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the association for computational linguistics*, 2:67–78, 2014.
- [78] Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*, 2022.
- [79] Lu Yuan, Dongdong Chen, Yi-Ling Chen, Noel Codella, Xiyang Dai, Jianfeng Gao, Houdong Hu, Xuedong Huang, Boxin Li, Chunyuan Li, et al. Florence: A new foundation model for computer vision. *arXiv preprint arXiv:2111.11432*, 2021.
- [80] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11975–11986, 2023.
- [81] Jingwei Zhang, Cheuk Ting Li, and Farzan Farnia. An interpretable evaluation of entropy-based novelty of generative models. *arXiv preprint arXiv:2402.17287*, 2024.
- [82] Jingwei Zhang, Mohammad Jalali, Cheuk Ting Li, and Farzan Farnia. Unveiling differences in generative models: A scalable differential clustering approach. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 8269–8278, 2025.
- [83] Yuhui Zhang, Elaine Sui, and Serena Yeung-Levy. Connect, collapse, corrupt: Learning cross-modal tasks with uni-modal data. *arXiv preprint arXiv:2401.08567*, 2024.
- [84] Taoxu Zhao, Meisi Li, Kehao Chen, Liye Wang, Xucheng Zhou, Kunal Chaturvedi, Mukesh Prasad, Ali Anaissi, and Ali Braytee. Enhancing sentiment analysis through multimodal fusion: A bert-dinov2 approach, 2025.

A Proofs

A.1 Proof of Proposition 1

To show the proposition, we only need to note the following about the inner product of the feature map $\phi_{\gamma_1, \gamma_2}$ for two samples x, y :

$$\begin{aligned}\phi_{\gamma_1, \gamma_2}(x)^\top \phi_{\gamma_1, \gamma_2}(y) &= \left(\phi_1(\gamma_1(x)) \otimes \phi_2(\gamma_2(x)) \right)^\top \left(\phi_1(\gamma_1(y)) \otimes \phi_2(\gamma_2(y)) \right) \\ &= \left(\phi_1(\gamma_1(x))^\top \phi_1(\gamma_1(y)) \right) \otimes \left(\phi_2(\gamma_2(x))^\top \phi_2(\gamma_2(y)) \right) \\ &= k_{\gamma_1}(x, y) \otimes k_{\gamma_2}(x, y) \\ &= k_{\gamma_1}(x, y) \cdot k_{\gamma_2}(x, y).\end{aligned}$$

A.2 Proof of Proposition 2

To show the proposition, we validate the claimed identities one by one. For the first equation, note that

$$\begin{aligned}\langle E_X(x), E_X(x') \rangle &= \left(\phi(\Psi_X(x)) \otimes \tilde{\phi}_{\gamma, X}(x) \right)^\top \left(\phi(\Psi_X(x')) \otimes \tilde{\phi}_{\gamma, X}(x') \right) \\ &= \left(\phi(\Psi_X(x))^\top \phi(\Psi_X(x')) \right) \otimes \left(\tilde{\phi}_{\gamma, X}(x)^\top \tilde{\phi}_{\gamma, X}(x') \right) \\ &= k(\Psi_X(x), \Psi_X(x')) \left(\tilde{\phi}_{\gamma, X}(x)^\top \tilde{\phi}_{\gamma, X}(x') \right) \\ &= k(\Psi_X(x), \Psi_X(x')) \left(\frac{1}{2} \cdot \frac{C}{d} \cdot 2d + \frac{1}{2} \cdot 2 \cdot \phi_{\gamma, X}(x)^\top \phi_{\gamma, X}(x') \right) \\ &= k(\Psi_X(x), \Psi_X(x')) \left(C + k(\gamma_X(x), \gamma_X(x')) \right)\end{aligned}$$

For the second identity, note that

$$\begin{aligned}\langle E_T(t), E_T(t') \rangle &= \left(\phi(\Psi_T(t)) \otimes \tilde{\phi}_{\gamma, T}(t) \right)^\top \left(\phi(\Psi_T(t')) \otimes \tilde{\phi}_{\gamma, T}(t') \right) \\ &= \left(\phi(\Psi_T(t))^\top \phi(\Psi_T(t')) \right) \otimes \left(\tilde{\phi}_{\gamma, T}(t)^\top \tilde{\phi}_{\gamma, T}(t') \right) \\ &= k(\Psi_T(t), \Psi_T(t')) \left(\tilde{\phi}_{\gamma, T}(t)^\top \tilde{\phi}_{\gamma, T}(t') \right) \\ &= k(\Psi_T(t), \Psi_T(t')) \left(\frac{C}{2d} \cdot 2d \right) \\ &= k(\Psi_T(t), \Psi_T(t')) \cdot C\end{aligned}$$

Finally, for the last identity, we can complete the proof as:

$$\begin{aligned}\langle E_X(x), E_T(t) \rangle &= \left(\phi(\Psi_X(x)) \otimes \tilde{\phi}_{\gamma, X}(x) \right)^\top \left(\phi(\Psi_T(t)) \otimes \tilde{\phi}_{\gamma, T}(t) \right) \\ &= \left(\phi(\Psi_X(x))^\top \phi(\Psi_T(t)) \right) \otimes \left(\tilde{\phi}_{\gamma, X}(x)^\top \tilde{\phi}_{\gamma, T}(t) \right) \\ &= k(\Psi_X(x), \Psi_T(t)) \left(\tilde{\phi}_{\gamma, X}(x)^\top \tilde{\phi}_{\gamma, T}(t) \right) \\ &= k(\Psi_X(x), \Psi_T(t)) \left(\frac{1}{2} \cdot \frac{C}{d} \cdot 2d \right) \\ &= k(\Psi_X(x), \Psi_T(t)) \cdot C.\end{aligned}$$

A.3 Proof of Theorem 1

To show this statement, we observe that the Kronecker product embedding can be written as:

$$\begin{aligned}
\psi_{1,2,X}(x_i)^\top \psi_{1,2,X}(x_j) &= \left(\psi_{1,X}(x_i) \otimes \psi_{2,X}(x_i) \right)^\top \left(\psi_{1,X}(x_j) \otimes \psi_{2,X}(x_j) \right) \\
&= \left(\psi_{1,X}(x_i)^\top \psi_{1,X}(x_j) \right) \otimes \left(\psi_{2,X}(x_i)^\top \psi_{2,X}(x_j) \right) \\
&= \left(\psi_{1,X}(x_i)^\top \psi_{1,X}(x_j) \right) \cdot \left(\psi_{2,X}(x_i)^\top \psi_{2,X}(x_j) \right) \\
&= \mathbb{E}_{\mathbf{u}_1} \left[\psi_{1,X}(x_i)^\top \mathbf{u}_1^\top \mathbf{u}_1 \psi_{1,X}(x_j) \right] \cdot \mathbb{E}_{\mathbf{u}_2} \left[\psi_{2,X}(x_i)^\top \mathbf{u}_2^\top \mathbf{u}_2 \psi_{2,X}(x_j) \right] \\
&= \mathbb{E}_{\mathbf{u}_1, \mathbf{u}_2} \left[\left(\psi_{1,X}(x_i)^\top \mathbf{u}_1^\top \mathbf{u}_1 \psi_{1,X}(x_j) \right) \cdot \left(\psi_{2,X}(x_i)^\top \mathbf{u}_2^\top \mathbf{u}_2 \psi_{2,X}(x_j) \right) \right]
\end{aligned}$$

where in the above $\mathbf{u}_1 \in \mathbb{R}^{d_1}$ and $\mathbf{u}_2 \in \mathbb{R}^{d_2}$ are independent random vectors with zero mean and covariance matrix $I_{d_1 \times d_1}$ and $I_{d_2 \times d_2}$. It can be seen that each row of the matrix U_1 and U_2 with uniformly-drawn entries over $[-\sqrt{3}, \sqrt{3}]$ satisfies this property. As a result, for every row $\mathbf{u}_{1,i}$ of U_1 and row $\mathbf{u}_{2,i}$ of U_2 (with randomly drawn entries with the specified distribution) we have:

$$\mathbb{E}_{\mathbf{u}_{1,i}, \mathbf{u}_{2,i}} \left[\tilde{\psi}_{X, \mathbf{u}_{1,i}, \mathbf{u}_{2,i}}(x_i)^\top \tilde{\psi}_{X, \mathbf{u}_{1,i}, \mathbf{u}_{2,i}}(x_j) \right] = \psi_{1,2,X}(x_i)^\top \psi_{1,2,X}(x_j)$$

Note that we have $\tilde{\psi}_X(x_i)^\top \tilde{\psi}_X(x_j) = \frac{1}{l} \sum_{i=1}^l \tilde{\psi}_{X, \mathbf{u}_{1,i}, \mathbf{u}_{2,i}}(x_i)^\top \tilde{\psi}_{X, \mathbf{u}_{1,i}, \mathbf{u}_{2,i}}(x_j)$. On the other hand, we know that each random variable in this empirical mean is bounded as $|\tilde{\psi}_{X, \mathbf{u}_{1,2,i}}(x_i)^\top \tilde{\psi}_{X, \mathbf{u}_{1,2,i}}(x_j)| \leq \|\tilde{\psi}_{X, \mathbf{u}_{1,2,i}}(x_i)\|_2 \|\tilde{\psi}_{X, \mathbf{u}_{1,2,i}}(x_j)\|_2 \leq B$. Therefore, we can apply Hoeffding's inequality to show that

$$\mathbb{P} \left(\left| \tilde{\psi}_X(x_i)^\top \tilde{\psi}_X(x_j) - \psi_{1,2,X}(x_i)^\top \psi_{1,2,X}(x_j) \right| \geq \epsilon \right) \leq 2 \exp \left(\frac{-l\epsilon^2}{2B^2} \right)$$

Setting $\delta = 2n^2 \exp \left(\frac{-l\epsilon^2}{2B^2} \right)$ which implies that $\epsilon = \sqrt{\frac{2B^2 \log(2n^2/\delta)}{l}}$, shows that for every $1 \leq i, j \leq n$ we have that

$$\mathbb{P} \left(\left| \tilde{\psi}_X(x_i)^\top \tilde{\psi}_X(x_j) - \psi_{1,2,X}(x_i)^\top \psi_{1,2,X}(x_j) \right| \geq \sqrt{\frac{2B^2 \log(2n^2/\delta)}{l}} \right) \leq \frac{\delta}{n^2}$$

Thus, applying the union bound to all n^2 pairs $(i, j) \in \{1, \dots, n\}^2$ shows that

$$\mathbb{P} \left(\forall i, j : \left| \tilde{\psi}_X(x_i)^\top \tilde{\psi}_X(x_j) - \psi_{1,2,X}(x_i)^\top \psi_{1,2,X}(x_j) \right| \leq \sqrt{\frac{2B^2 \log(2n^2/\delta)}{l}} \right) \geq 1 - \delta.$$

We can similarly prove the above result for the other modality:

$$\mathbb{P} \left(\forall i, j : \left| \tilde{\psi}_T(t_i)^\top \tilde{\psi}_T(t_j) - \psi_{1,2,T}(t_i)^\top \psi_{1,2,T}(t_j) \right| \leq \sqrt{\frac{2B^2 \log(2n^2/\delta)}{l}} \right) \geq 1 - \delta.$$

This completes the proof.

A.4 Extending KrossFuse to Infinite-dimension Shift-Invariant Kernels

As we discussed in the main text, a feasible application of KrossFuse requires a feature map $\phi : \mathcal{X} \rightarrow \mathbb{R}^s$ mapping to a finite-dimensional space with $s < \infty$. However, this assumption does not apply to popular shift-invariant kernels including the Gaussian (RBF) kernel $k_{\text{gaussian}(\sigma)}(x, y) = \exp(-\|x - y\|_2^2 / 2\sigma^2)$ and the Laplace kernel $k_{\text{laplace}(\eta)}(x, y) = \exp(-\|x - y\|_1 / \eta)$.

To extend the KrossFuse application to a general shift-invariant kernel $k(x, y) = \kappa(x - y)$ without any assumption on the finiteness of its feature map, we propose the application of the random Fourier feature (RFF) framework in [61]. Note that if a shift-invariant map $k(x, y) = \kappa(x - y)$ for $\kappa : \mathcal{X} \rightarrow \mathbb{R}$ satisfies the positive semi-definite property of a kernel function, Bochner's theorem shows that the Fourier transform $\hat{\kappa} : \mathbb{R}^d \rightarrow \mathbb{R}$ of κ will take real non-negative values everywhere, $\hat{\kappa}(\omega) \geq 0, \forall \omega$. Note that we define the Fourier transform as follows where $\langle \omega, x \rangle$ denotes the standard inner product in the $\mathcal{X}$ space.

$$\hat{\kappa}(\omega) = \frac{1}{(2\pi)^d} \int_{\mathcal{X}} \kappa(x) \exp(-i\langle \omega, x \rangle) dx. \quad (8)$$

Given the above definition, it can be seen that the synthesis equation implies $\kappa(0) = \int_{\omega} \hat{\kappa}(\omega) d\omega$, which means $\hat{\kappa}$ is a valid probability density function (PDF) for every normalized shift-invariant kernel k satisfying $k(x, x) = \kappa(0) = 1$.

Therefore, the synthesis equation shows that

$$\begin{aligned}
k(x, y) &= \kappa(x - y) \\
&= \int_{\mathcal{X}} \hat{\kappa}(\omega) \exp(i\omega^\top (x - y)) d\omega \\
&= \int_{\mathcal{X}} \hat{\kappa}(\omega) \cos(\omega^\top (x - y)) d\omega \\
&= \mathbb{E}_{\omega \sim \hat{\kappa}} [\cos(\omega^\top (x - y))] \\
&= \mathbb{E}_{\omega \sim \hat{\kappa}} [\cos(\omega^\top x) \cos(\omega^\top y) + \sin(\omega^\top x) \sin(\omega^\top y)] \\
&= \mathbb{E}_{\omega \sim \hat{\kappa}} [\cos(\omega^\top x), \sin(\omega^\top x)]^\top [\cos(\omega^\top y), \sin(\omega^\top y)]
\end{aligned}$$

Note that given a single embedding $\gamma : \mathcal{X} \rightarrow \mathbb{R}^d$, the above characterization leads to the standard RFF framework, where for a RFF feature size $r \in \mathbb{N}$, we draw IID random samples $\omega_1, \dots, \omega_r \sim \hat{\kappa}$ and define the following RFF proxy map $\phi_r : \mathbb{R}^d \rightarrow \mathbb{R}^{2r}$:

$$\phi_r(z) = \frac{1}{\sqrt{r}} [\cos(\omega_1^\top z), \sin(\omega_1^\top z), \dots, \cos(\omega_r^\top z), \sin(\omega_r^\top z)] \quad (9)$$

However, the application of the RFF framework to the Kronecker product of the embeddings γ_1 and γ_2 remains unclear. In this work, we propose a joint sampling of RFF features for the Kronecker product of embeddings. More specifically, suppose we consider shift-invariant kernel functions $k_1(x, x') = \kappa_1(x - x')$ for embedding map γ_1 and $k_2(x, x') = \kappa_2(x - x')$ for embedding map γ_2 . Then, we consider the joint probability density function $M(\omega_1, \omega_2) = \hat{\kappa}_1(\omega_1) \cdot \hat{\kappa}_2(\omega_2)$ for independent variables ω_1, ω_2 .

For applying the RFF framework to the Kronecker product of the embeddings under shift-invariant kernels, we propose IID sampling $(\omega_1^{(i)}, \omega_2^{(i)}) \sim M$, i.e., we draw the r samples jointly, instead of generating them separately for each samples and consider the grid-based pairing of the drawn samples. Given the r drawn samples $(\omega_1^{(i)}, \omega_2^{(i)})_{i=1}^r$, we define the following joint RFF feature map:

$$\begin{aligned}
\hat{\phi}_r(z_1, z_2) &= \frac{1}{\sqrt{r}} [\cos(z_1^\top \omega_1^{(1)} + z_2^\top \omega_2^{(1)}), \sin(z_1^\top \omega_1^{(1)} + z_2^\top \omega_2^{(1)}), \\
&\quad \dots, \cos(z_1^\top \omega_1^{(r)} + z_2^\top \omega_2^{(r)}), \sin(z_1^\top \omega_1^{(r)} + z_2^\top \omega_2^{(r)})] \quad (10)
\end{aligned}$$

Similar to the standard case in the single embedding application of Fourier features, we can prove the following proposition:

Theorem 2. Consider two embedding maps $\gamma_1 : \mathcal{X} \rightarrow \mathbb{R}^{d_1}$ and $\gamma_2 : \mathcal{X} \rightarrow \mathbb{R}^{d_2}$, applied with shift-invariant kernel similarity functions $k_1 : \mathbb{R}^{d_1} \times \mathbb{R}^{d_1} \rightarrow \mathbb{R}$ and $k_2 : \mathbb{R}^{d_2} \times \mathbb{R}^{d_2} \rightarrow \mathbb{R}$, respectively. Assume $(\omega_1^{(i)}, \omega_2^{(i)})_{i=1}^r$ are drawn independently according to $\hat{\kappa}_1 \times \hat{\kappa}_2$, and define the proxy feature map $\hat{\phi}_r : \mathbb{R}^{d_1} \times \mathbb{R}^{d_2} \rightarrow \mathbb{R}^{2r}$ as in (10). Then, for every $x, y \in \mathcal{X}$ and every $\delta > 0$, the following holds with probability at least $1 - \delta$:

$$|k_1(\gamma_1(x), \gamma_1(y)) \cdot k_2(\gamma_2(x), \gamma_2(y)) - \langle \phi_r(\gamma_1(x), \gamma_2(x)), \phi_r(\gamma_1(y), \gamma_2(y)) \rangle| \leq \sqrt{\frac{2 \log(2/\delta)}{r}}$$

Proof. Given that $k_1(x, y) = \kappa_1(x - y)$ is a shift-invariant kernel that satisfies $\kappa_1(0) = 1$, we can deduce that

$$k_1(\gamma_1(x), \gamma_1(y)) = \mathbb{E}_{\omega_1 \sim \hat{\kappa}_1} [\cos(\omega_1^\top (\gamma_1(x) - \gamma_1(y)))]$$

Similarly, we can observe that $k_2(\gamma_2(x), \gamma_2(y)) = \mathbb{E}_{\omega_2 \sim \hat{\kappa}_2} [\cos(\omega_2^\top (\gamma_2(x) - \gamma_2(y)))]$, which results in

$$\begin{aligned}
& k_1(\gamma_1(x), \gamma_1(y)) \cdot k_2(\gamma_2(x), \gamma_2(y)) \\
&= \mathbb{E}_{\omega_1 \sim \hat{\kappa}_1} [\exp(i\omega_1^\top (\gamma_1(x) - \gamma_1(y)))] \cdot \mathbb{E}_{\omega_2 \sim \hat{\kappa}_2} [\exp(i\omega_2^\top (\gamma_2(x) - \gamma_2(y)))] \\
&= \mathbb{E}_{(\omega_1, \omega_2) \sim \hat{\kappa}_1 \times \hat{\kappa}_2} [\exp(i\omega_1^\top (\gamma_1(x) - \gamma_1(y))) \cdot \exp(i\omega_2^\top (\gamma_2(x) - \gamma_2(y)))] \\
&= \mathbb{E}_{(\omega_1, \omega_2) \sim \hat{\kappa}_1 \times \hat{\kappa}_2} [\exp(i\omega_1^\top (\gamma_1(x) - \gamma_1(y)) + i\omega_2^\top (\gamma_2(x) - \gamma_2(y)))] \\
&= \mathbb{E}_{(\omega_1, \omega_2) \sim \hat{\kappa}_1 \times \hat{\kappa}_2} [\exp(i((\omega_1^\top \gamma_1(x) + \omega_2^\top \gamma_2(x)) - (\omega_1^\top \gamma_1(y) + \omega_2^\top \gamma_2(y))))] \\
&= \mathbb{E}_{(\omega_1, \omega_2) \sim \hat{\kappa}_1 \times \hat{\kappa}_2} [\cos((\omega_1^\top \gamma_1(x) + \omega_2^\top \gamma_2(x)) - (\omega_1^\top \gamma_1(y) + \omega_2^\top \gamma_2(y)))] \\
&\quad + i \cdot \mathbb{E}_{(\omega_1, \omega_2) \sim \hat{\kappa}_1 \times \hat{\kappa}_2} [\sin((\omega_1^\top \gamma_1(x) + \omega_2^\top \gamma_2(x)) - (\omega_1^\top \gamma_1(y) + \omega_2^\top \gamma_2(y)))] \\
&= \mathbb{E}_{(\omega_1, \omega_2) \sim \hat{\kappa}_1 \times \hat{\kappa}_2} [\cos((\omega_1^\top \gamma_1(x) + \omega_2^\top \gamma_2(x)) - (\omega_1^\top \gamma_1(y) + \omega_2^\top \gamma_2(y)))] \\
&= \mathbb{E}_{(\omega_1, \omega_2) \sim \hat{\kappa}_1 \times \hat{\kappa}_2} [\cos(\omega_1^\top \gamma_1(x) + \omega_2^\top \gamma_2(x)) \cdot \cos(\omega_1^\top \gamma_1(y) + \omega_2^\top \gamma_2(y)) \\
&\quad + \sin(\omega_1^\top \gamma_1(x) + \omega_2^\top \gamma_2(x)) \cdot \sin(\omega_1^\top \gamma_1(y) + \omega_2^\top \gamma_2(y))].
\end{aligned}$$

Therefore, since $|\cos(\cdot)| \leq 1$, the application of Hoeffding's inequality implies that:

$$\begin{aligned}
& \mathbb{P}\left(\left|k_1(\gamma_1(x), \gamma_1(y)) \cdot k_2(\gamma_2(x), \gamma_2(y)) - \langle \phi_r(\gamma_1(x), \gamma_2(x)), \phi_r(\gamma_1(y), \gamma_2(y)) \rangle\right| \geq \epsilon\right) \\
&= \mathbb{P}\left(\left|\mathbb{E}_{(\omega_1, \omega_2) \sim \hat{\kappa}_1 \times \hat{\kappa}_2} [\cos((\omega_1^\top \gamma_1(x) + \omega_2^\top \gamma_2(x)) - (\omega_1^\top \gamma_1(y) + \omega_2^\top \gamma_2(y)))]\right.\right. \\
&\quad \left.- \frac{1}{r} \sum_{j=1}^r [\cos((\omega_1^{(j)\top} \gamma_1(x) + \omega_2^{(j)\top} \gamma_2(x)) - (\omega_1^{(j)\top} \gamma_1(y) + \omega_2^{(j)\top} \gamma_2(y)))]\right| \geq \epsilon\right) \\
&\leq 2 \exp\left(\frac{-2r\epsilon^2}{4}\right) \\
&= 2 \exp\left(\frac{-r\epsilon^2}{2}\right)
\end{aligned}$$

If we let $\delta = 2 \exp(\frac{-r\epsilon^2}{2})$, i.e., $\epsilon = \sqrt{\frac{2 \log(2/\delta)}{r}}$, then we can equivalently write

$$\begin{aligned}
& \mathbb{P}\left(\left|k_1(\gamma_1(x), \gamma_1(y)) \cdot k_2(\gamma_2(x), \gamma_2(y)) - \langle \phi_r(\gamma_1(x), \gamma_2(x)), \phi_r(\gamma_1(y), \gamma_2(y)) \rangle\right|\right. \\
&\quad \left.\geq \sqrt{\frac{2 \log(2/\delta)}{r}}\right) \leq \delta
\end{aligned}$$

which completes the proof. $\square$

Following Theorem 2, we can extend the random projection fusion of uni-modal embeddings to the settings with shift-invariant kernels that possess an infinite-dimensional feature map. In this case, the fusion will consider the jointly-drawn RFFs according to (10) of embeddings γ_1 with a shift-invariant kernel k_1 and γ_2 with a shift-invariant kernel k_2 . This $2r$ -dimensional vector will be the proxy fusion of the embedding maps (using the shift-invariant kernel similarity functions k_1, k_2). Theorem 1 also proves that the proxy kernel function of this RFF-based fusion is, with high probability, close to the product of marginal kernel functions that is the kernel function of the Kronecker product of the embeddings.

To further extend this discussion to the KrossFuse fusion of the cross-modal $\gamma = (\gamma_X, \gamma_T)$ and uni-modal ψ_X , we derive the following formulation. We first generate the RFF features in (10) to obtain the $2r$ -dimensional map $\psi_r(x)$ for the shared modality. Extending the RP-KrossFuse

definition to shift-invariant kernels, we obtain the following fused embeddings $\tilde{\psi}_X : \mathcal{X} \rightarrow \mathbb{R}^{8r}$ and $\tilde{\psi}_T : \mathcal{T} \rightarrow \mathbb{R}^{8r}$ for the uni-modal embedding ψ_X functioning only the modality X and cross-modal embedding $\gamma = (\gamma_X, \gamma_T)$:

$$\begin{aligned} \tilde{\psi}_X(x) := & \frac{1}{\sqrt{2r}} \left[\sqrt{C} \cos(\psi_X(x)^\top \omega_2^{(1)}) + \cos(\gamma_X(x)^\top \omega_1^{(1)} + \psi_X(x)^\top \omega_2^{(1)}), \right. \\ & \sqrt{C} \cos(\psi_X(x)^\top \omega_2^{(1)}) - \cos(\gamma_X(x)^\top \omega_1^{(1)} + \psi_X(x)^\top \omega_2^{(1)}), \\ & \sqrt{C} \cos(\psi_X(x)^\top \omega_2^{(1)}) + \sin(\gamma_X(x)^\top \omega_1^{(1)} + \psi_X(x)^\top \omega_2^{(1)}), \\ & \sqrt{C} \cos(\psi_X(x)^\top \omega_2^{(1)}) - \sin(\gamma_X(x)^\top \omega_1^{(1)} + \psi_X(x)^\top \omega_2^{(1)}), \\ & \sqrt{C} \sin(\psi_X(x)^\top \omega_2^{(1)}) + \cos(\gamma_X(x)^\top \omega_1^{(1)} + \psi_X(x)^\top \omega_2^{(1)}), \\ & \sqrt{C} \sin(\psi_X(x)^\top \omega_2^{(1)}) - \cos(\gamma_X(x)^\top \omega_1^{(1)} + \psi_X(x)^\top \omega_2^{(1)}), \\ & \sqrt{C} \sin(\psi_X(x)^\top \omega_2^{(1)}) + \sin(\gamma_X(x)^\top \omega_1^{(1)} + \psi_X(x)^\top \omega_2^{(1)}), \\ & \sqrt{C} \sin(\psi_X(x)^\top \omega_2^{(1)}) - \sin(\gamma_X(x)^\top \omega_1^{(1)} + \psi_X(x)^\top \omega_2^{(1)}) \\ & , \dots, \\ & \sqrt{C} \cos(\psi_X(x)^\top \omega_2^{(r)}) + \cos(\gamma_X(x)^\top \omega_1^{(r)} + \psi_X(x)^\top \omega_2^{(r)}), \\ & \sqrt{C} \cos(\psi_X(x)^\top \omega_2^{(r)}) - \cos(\gamma_X(x)^\top \omega_1^{(r)} + \psi_X(x)^\top \omega_2^{(r)}), \\ & \sqrt{C} \cos(\psi_X(x)^\top \omega_2^{(r)}) + \sin(\gamma_X(x)^\top \omega_1^{(r)} + \psi_X(x)^\top \omega_2^{(r)}), \\ & \sqrt{C} \cos(\psi_X(x)^\top \omega_2^{(r)}) - \sin(\gamma_X(x)^\top \omega_1^{(r)} + \psi_X(x)^\top \omega_2^{(r)}), \\ & \sqrt{C} \sin(\psi_X(x)^\top \omega_2^{(r)}) + \cos(\gamma_X(x)^\top \omega_1^{(r)} + \psi_X(x)^\top \omega_2^{(r)}), \\ & \sqrt{C} \sin(\psi_X(x)^\top \omega_2^{(r)}) - \cos(\gamma_X(x)^\top \omega_1^{(r)} + \psi_X(x)^\top \omega_2^{(r)}), \\ & \sqrt{C} \sin(\psi_X(x)^\top \omega_2^{(r)}) + \sin(\gamma_X(x)^\top \omega_1^{(r)} + \psi_X(x)^\top \omega_2^{(r)}), \\ & \left. \sqrt{C} \sin(\psi_X(x)^\top \omega_2^{(r)}) - \sin(\gamma_X(x)^\top \omega_1^{(r)} + \psi_X(x)^\top \omega_2^{(r)}) \right] \end{aligned}$$

and

$$\begin{aligned} \tilde{\psi}_T(t) := & \frac{1}{\sqrt{2r}} \left[\sqrt{C} + \cos(\gamma_T(t)^\top \omega_1^{(1)}), \sqrt{C} - \cos(\gamma_T(t)^\top \omega_1^{(1)}), \sqrt{C} + \sin(\gamma_T(t)^\top \omega_1^{(1)}) \right. \\ & , \sqrt{C} - \sin(\gamma_T(t)^\top \omega_1^{(1)}), \\ & \cos(\gamma_T(t)^\top \omega_1^{(1)}), -\cos(\gamma_T(t)^\top \omega_1^{(1)}), \sin(\gamma_T(t)^\top \omega_1^{(1)}), -\sin(\gamma_T(t)^\top \omega_1^{(1)}) \\ & , \dots, \\ & \sqrt{C} + \cos(\gamma_T(t)^\top \omega_1^{(r)}), \sqrt{C} - \cos(\gamma_T(t)^\top \omega_1^{(r)}), \sqrt{C} + \sin(\gamma_T(t)^\top \omega_1^{(r)}) \\ & \sqrt{C} - \sin(\gamma_T(t)^\top \omega_1^{(r)}), \\ & \left. \cos(\gamma_T(t)^\top \omega_1^{(r)}), -\cos(\gamma_T(t)^\top \omega_1^{(r)}), \sin(\gamma_T(t)^\top \omega_1^{(r)}), -\sin(\gamma_T(t)^\top \omega_1^{(r)}) \right] \end{aligned}$$

A.5 Algorithm of Fusing Cross-modal Embeddings and Uni-modal Embeddings

Algorithm 2 Kernel Feature Fusion of Cross-modal Embeddings and Uni-modal Embeddings

```
1: Input: Image samples  $\{x_i\}_{i=1}^{N_x}$ , Text samples  $\{y_j\}_{j=1}^{N_y}$ , Cross-modal encoder  $\gamma_1$ , Uni-image  
   encoder  $\gamma_2$  and Uni-text encoder  $\gamma_3$ , Kernel maps  $\phi_1, \phi_2, \phi_3$ , Constant  $c$ , Projected dim  $l$   
2:  $U_i \sim \text{Uniform}[-\sqrt{3}, \sqrt{3}]^{d_{\phi_i} \times l} / \sqrt{l}$ , for  $i = 1, 2, 3$ .  
3: Initialize  $Z^{\text{img}} \in \mathbb{R}^{N_x \times l}$ ,  $Z^{\text{text}} \in \mathbb{R}^{N_y \times l}$   
4: for batch  $\mathcal{B}_x$  in  $\{x_i\}$  do ▷ Process Image Samples  
5:    $\psi_{1,\mathcal{B}_x} \leftarrow \phi_1(\gamma_1(\mathcal{B}_x))$ ,  $\psi_{2,\mathcal{B}_x} \leftarrow \phi_2(\gamma_2(\mathcal{B}_x))$ ,  $\psi_{3,\mathcal{B}_x} \leftarrow c \cdot \mathbf{1}_{|\mathcal{B}_x| \times 2d_{\phi_3}}$   
6:    $\psi'_{2,\mathcal{B}_x} \leftarrow \text{concat}(c\mathbf{1} + \psi_{2,\mathcal{B}_x}, c\mathbf{1} - \psi_{2,\mathcal{B}_x})$   
7:    $Z^{\text{img}}_{\mathcal{B}_x} \leftarrow (\psi_{1,\mathcal{B}_x} U_1) \odot (\psi'_{2,\mathcal{B}_x} U_2) \odot (\psi_{3,\mathcal{B}_x} U_3)$   
8: end for  
9: for batch  $\mathcal{B}_y$  in  $\{y_j\}$  do ▷ Process Text Samples  
10:   $\psi_{1,\mathcal{B}_y} \leftarrow \phi_1(\gamma_1(\mathcal{B}_y))$ ,  $\psi_{2,\mathcal{B}_y} \leftarrow c \cdot \mathbf{1}_{|\mathcal{B}_y| \times 2d_{\phi_2}}$ ,  $\psi_{3,\mathcal{B}_y} \leftarrow \phi_3(\gamma_3(\mathcal{B}_y))$   
11:   $\psi'_{3,\mathcal{B}_y} \leftarrow \text{concat}(c\mathbf{1} + \psi_{3,\mathcal{B}_y}, c\mathbf{1} - \psi_{3,\mathcal{B}_y})$   
12:   $Z^{\text{text}}_{\mathcal{B}_y} \leftarrow (\psi_{1,\mathcal{B}_y} U_1) \odot (\psi_{2,\mathcal{B}_y} U_2) \odot (\psi'_{3,\mathcal{B}_y} U_3)$   
13: end for  
14: Return  $Z^{\text{img}}, Z^{\text{text}}$ 
```

B Implementation Details

B.1 Experiments Setup

We tested several pre-trained embeddings in our experiments, including multiple variants of CLIP [60], DINOv2 [53], Unicom [3], Sroberta [62], E5 [74], and Siglip [80]. Unless otherwise specified, we primarily report results in the main text using standard CLIP(ViT-B/32), DINOv2(ViT-B/14), and Sroberta. The projection matrices U_i we generate once using the uniform distribution over $[-\sqrt{3}, \sqrt{3}]$, that has unit variance. The projection dimension l is 3000 for all datasets except 5000 in the case of ImageNet. The parameter C has been determined using cross-validation over $\{10^{-3}, 10^{-2}, \dots, 10^3\}$. All experiments were run on 2 RTX-4090 GPUs.

B.2 Baselines Setup

Our experiments analyze the fusion of cross-modal and uni-modal representations where we utilize the pretrained expert encoders without additional fine-tuning. In our numerical analysis, we consider the last-layer output for every attempted encoder. To the best of our knowledge, the task of fusing cross-modal and uni-modal embeddings has not been exclusively analyzed in the literature. Therefore, we emphasize that the baseline methods discussed in our analysis are fusion methods proposed for two unimodal embeddings, which cannot be applied to the zero-shot classification tasks.

The baseline methods in our analysis are: (1) *Kronecker Product of Marginal Random Projection (KPoMRP)*. This baseline applies an independent random projection separately to each embedding output, followed by a Kronecker product to obtain the fused representation. Note that our proposed RP-KrossFuse functions differently and samples the random features jointly for the output of the two embeddings. (2) *Gated Fusion*. This baseline represents the application of the Mixture-of-Experts (MoE) method discussed in [66], where a gating mechanism dynamically controls the contribution of different feature sources. Specifically, each feature is first passed to an MLP layer, then the outputs are passed through a sigmoid gating function to compute a dynamic fusion weight. (3) *Attentional Fusion*. The baseline follows the self-attentional fusion framework proposed in [84]. Projected features from each encoder are concatenated and passed through a self-attention module to dynamically aggregate information. (4) *COMM*. This baseline follows the fusion framework proposed in [33]. They employ an MLP layer to project the features of DINOv2 and concatenate the output features with that of CLIP.

C Additional Experimental Results

C.1 Unimodal Clustering

We present the complete set of our numerical results of kernel-based clustering on the image datasets: CUB-200-2011 [73], Oxford Flowers [51], DTD [11], Image-Woof [23] consisting of ten dog breeds from ImageNet-1K [13], GTSRB [69] and typographic attack images by introducing mislabeled red text into 10 ImageNet subclasses following the reference [47]. Table 4 reports the clustering performance scores of different methods on six image datasets, evaluated by Normalized Mutual Information [48] (NMI), Adjusted Mutual Information [71] (AMI), and Adjusted Rand Index [27] (ARI). Across all datasets and metrics, KrossFuse consistently performs better than individual CLIP and DINOv2, reaching the highest scores across the datasets. Notably, KrossFuse shows improvements on known challenging dataset cases such as Typo-Attacked ImageNet, ImageNet-Dogs and DTD, indicating its capability in capturing discriminative features for clustering tasks. The following figures further illustrate the detailed clustering results on these datasets. On top of it, we visualize the kernel matrices, as well as the distribution of the embeddings using t-SNE and UMAP, providing qualitative insights into the effectiveness of each method in separating different classes.

Table 4

Metric	Method	CUB200	Flowers102	DTD	ImageNet-Dogs	GTSRB	Typo-Attacked ImageNet
NMI	CLIP	63.2	80.0	50.9	49.7	49.7	20.1
	DINOv2	85.2	98.7	60.5	86.7	40.2	81.9
	KrossFuse	85.6	99.1	62.9	88.3	50.0	87.4
AMI	CLIP	45.7	73.2	47.7	49.2	46.4	19.3
	DINOv2	78.3	98.2	57.8	86.6	36.2	81.7
	KrossFuse	79.0	98.8	60.4	88.2	46.7	87.3
ARI	CLIP	21.1	55.5	27.2	37.1	18.6	10.4
	DINOv2	55.9	94.6	28.1	84.8	12.3	65.9
	KrossFuse	56.3	97.0	36.4	86.3	19.5	79.6

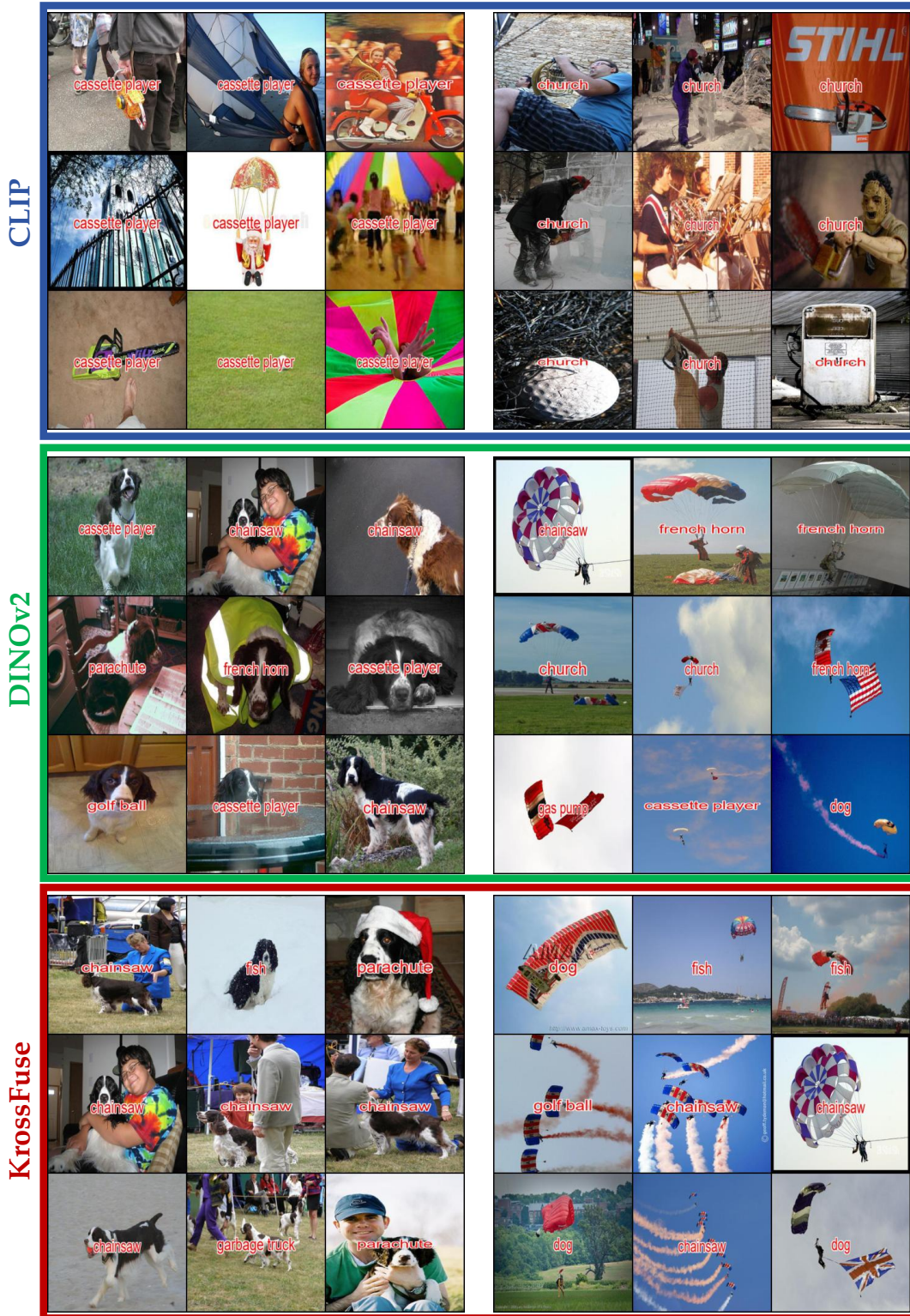


Figure 7: Clustering results of CLIP, DINOv2 and KrossFuse embeddings for typographic attacked images (red text is the misleading labels that simulate the attack) from 10 ImageNet classes. KrossFuse could cluster the attacked image classes like DINOv2 while CLIP is misled by text.

Typo-attacked ImageNet Dataset

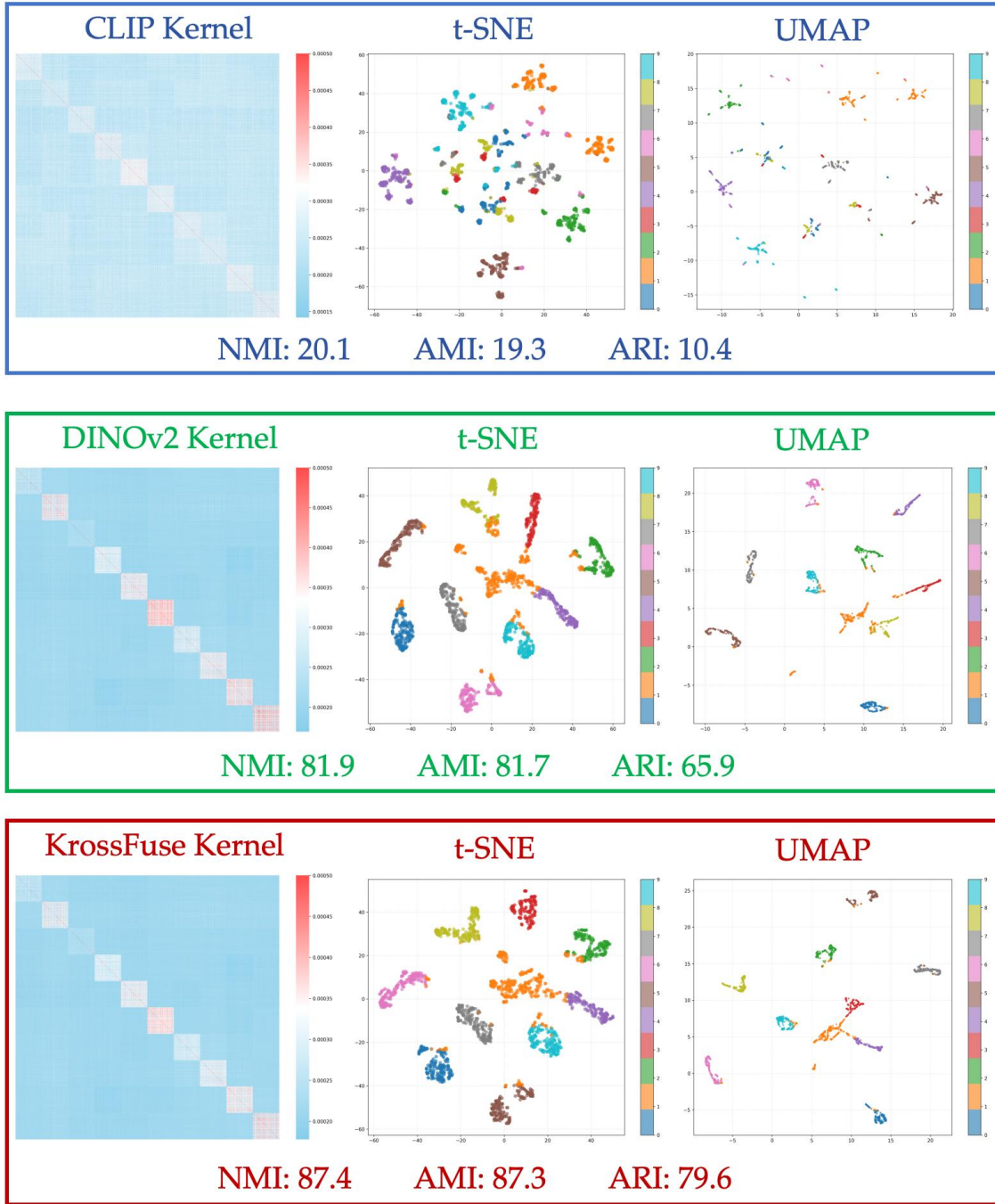


Figure 8: Comparison among CLIP, DINOv2 and KrossFuse embeddings for typographic attacked images from 10 ImageNet classes. (Left) Heatmaps of RBF kernel similarity matrices, (Middle) t-SNE visualization, (Right) UMAP visualization. KrossFuse could cluster the attacked image classes like DINOv2 while CLIP is misled by text.



Figure 9: Clustering results of CLIP, DINOv2 and KrossFuse embeddings for GTSRB dataset with eight clusters. KrossFuse could cluster the eight image classes like CLIP while DINOv2 can't distinguish them.

GTSRB Dataset

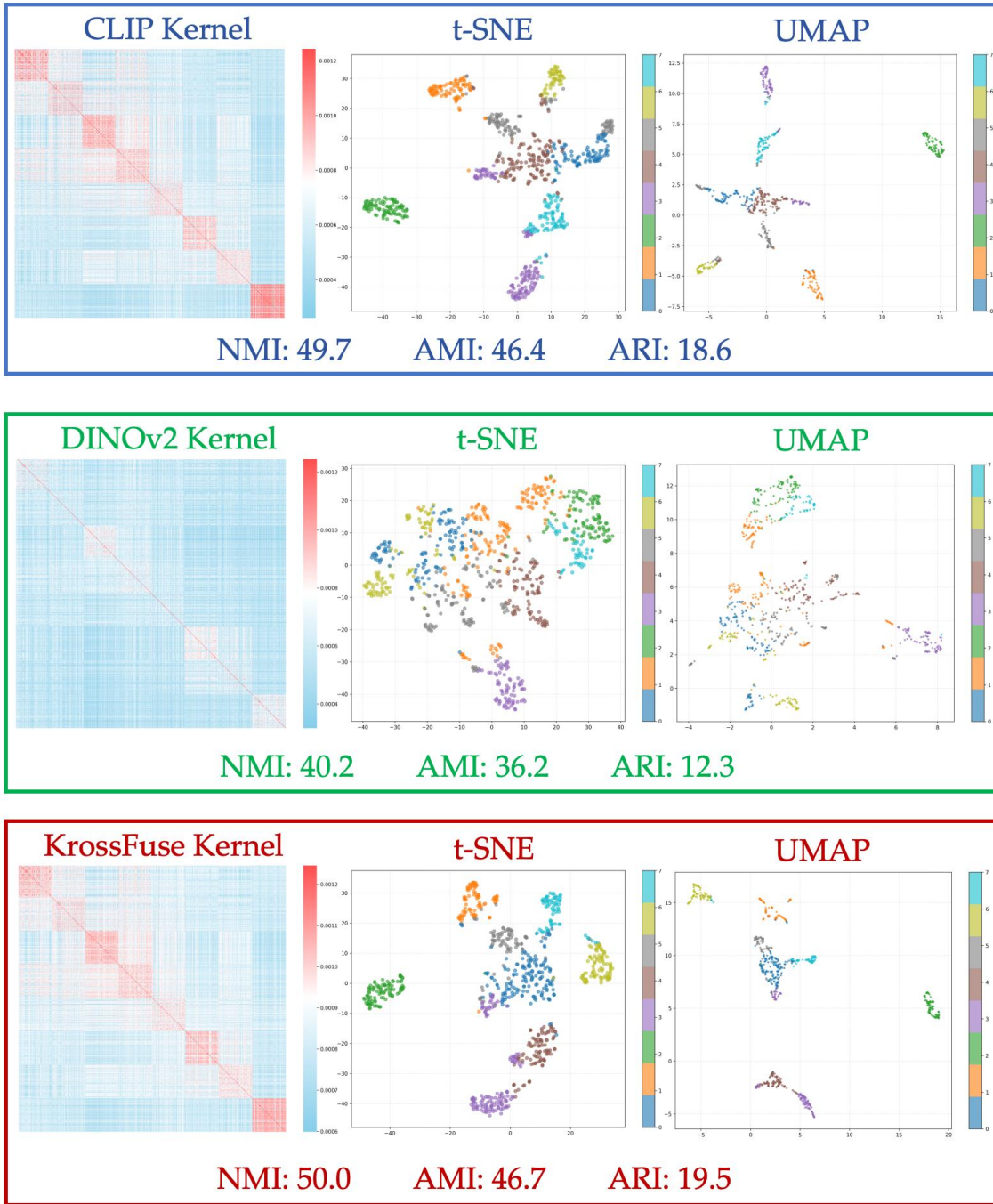


Figure 10: Comparison among CLIP, DINOv2 and KrossFuse embeddings for GTSRB dataset with eight clusters. (Left) Heatmaps of RBF kernel similarity matrices, (Middle) t-SNE visualization, (Right) UMAP visualization. KrossFuse could cluster the eight image classes like CLIP while DINOv2 can't distinguish all of them.



Figure 11: Clustering results of CLIP, DINOv2 and KrossFuse embeddings for ImageNet-dog breeds dataset. KrossFuse could cluster them like DINOv2 while CLIP can't distinguish all of them.

ImageNet-Dogs Dataset

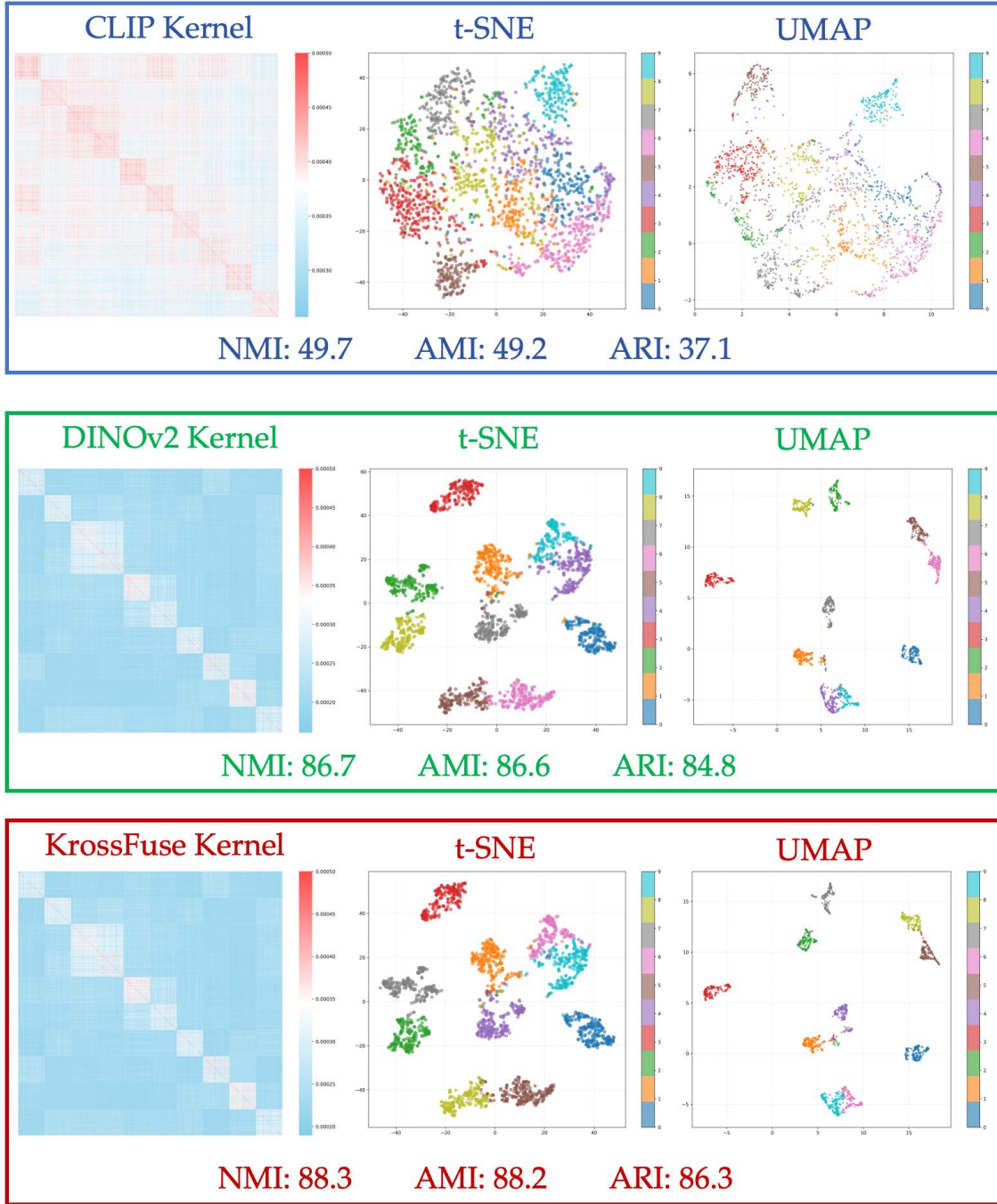


Figure 12: Comparison among CLIP, DINOv2 and KrossFuse embeddings for ImageNet-dog breeds dataset. (Left) Heatmaps of RBF kernel similarity matrices, (Middle) t-SNE visualization, (Right) UMAP visualization. KrossFuse could cluster the different dog classes like DINOv2 while CLIP can't distinguish all of them.



Figure 13: Clustering results of CLIP, DINOv2 and KrossFuse embeddings for DTD dataset. KrossFuse could cluster them like DINOv2 while CLIP can't distinguish all of them.

DTD Dataset

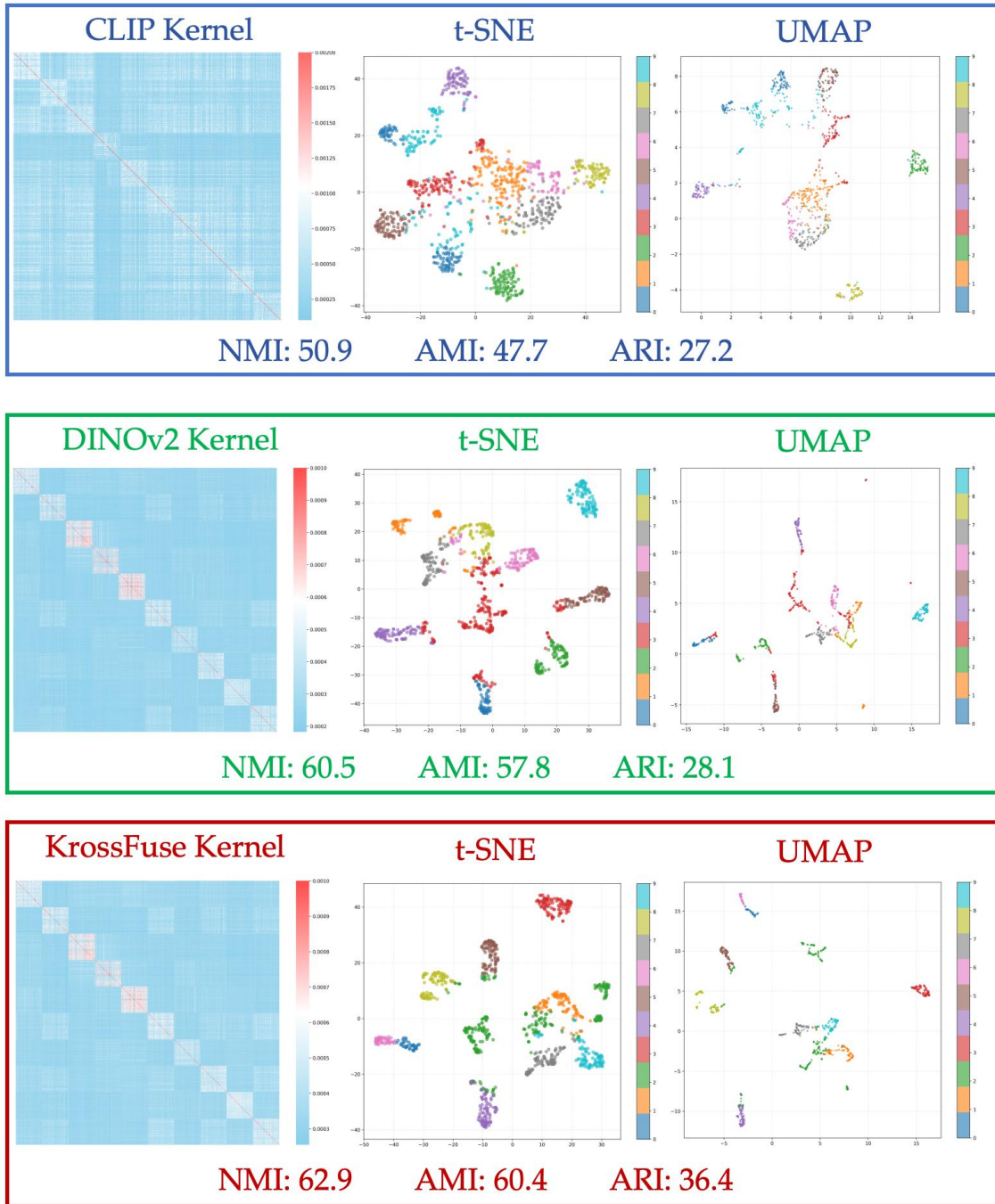


Figure 14: Comparison among CLIP, DINOv2 and KrossFuse embeddings for DTD dataset. (Left) Heatmaps of RBF kernel similarity matrices, (Middle) t-SNE visualization, (Right) UMAP visualization. KrossFuse could cluster the different texture classes like DINOv2 while CLIP can't distinguish all of them.



Figure 15: Clustering results of CLIP, DINOv2 and KrossFuse embeddings for CUB200 dataset. KrossFuse could cluster them like DINOv2 while CLIP can't distinguish all of them.

CUB200 Dataset

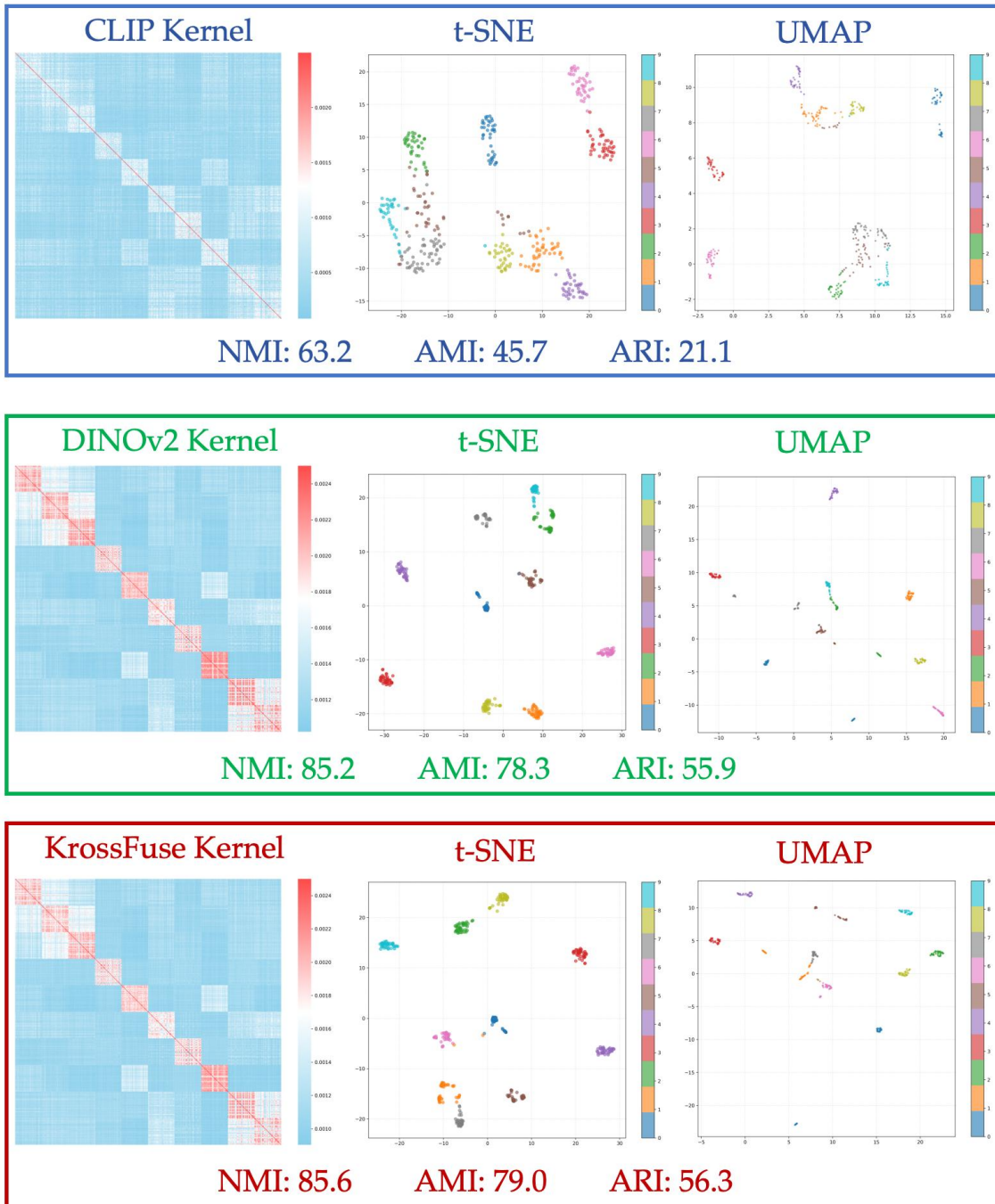


Figure 16: Comparison among CLIP, DINOv2 and KrossFuse embeddings for CUB200 dataset. (Left) Heatmaps of RBF kernel similarity matrices, (Middle) t-SNE visualization, (Right) UMAP visualization. KrossFuse could cluster the different bird classes like DINOv2 while CLIP can't distinguish all of them.



Figure 17: Clustering results of CLIP, DINOv2 and KrossFuse embeddings for Flowers102 dataset. KrossFuse could cluster them like DINOv2 while CLIP can't distinguish all of them.

Flowers Dataset

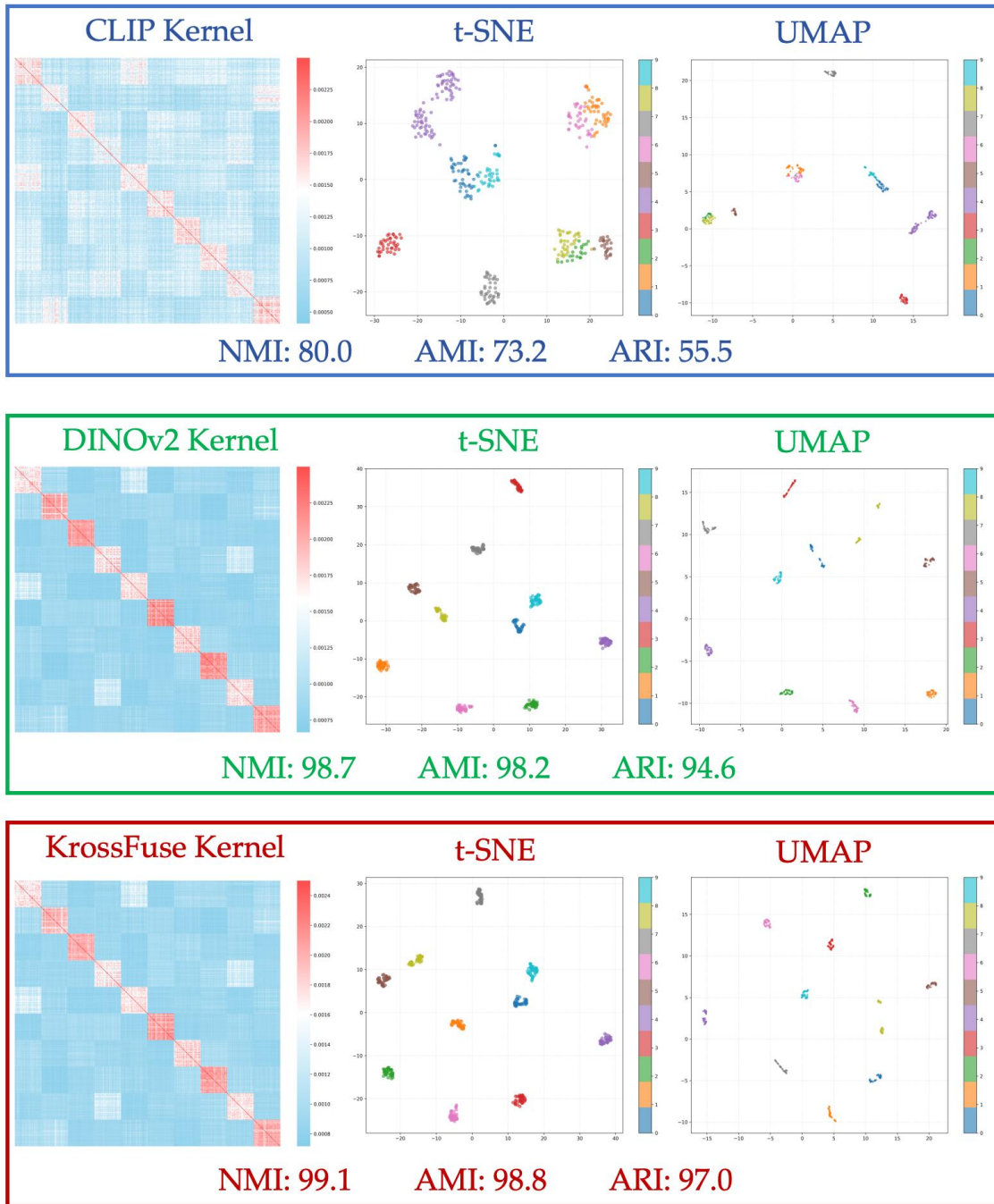


Figure 18: Comparison among CLIP, DINOv2 and KrossFuse embeddings for Flowers102 dataset. (Left) Heatmaps of RBF kernel similarity matrices, (Middle) t-SNE visualization, (Right) UMAP visualization. KrossFuse could cluster the different flower classes like DINOv2 while CLIP can't distinguish all of them.

C.2 Cross-modal Clustering

While our method achieves comparable performance to CLIP in zero shot image-text retrieval tasks, we observed an interesting phenomenon in cross-modal clustering: despite normalization to the unit hypersphere, image and text embeddings tend to form separate clusters due to the modality gap [41, 83]. This separation persists even when both modalities represent semantically identical concepts, indicating a systematic misalignment in the embedding spaces³. To address this challenge, we propose a learned unitary transformation applied to the normalized text embeddings. This rotation operation preserves the geometric structure within each modality while aligning the text embedding space with the image embedding space. Our approach ensures that the semantic relationships are maintained across modalities, enhancing the model’s ability to perform cross-modal tasks effectively. We validated our method on the MSCOCO dataset, where we applied t-SNE visualization to demonstrate the elimination of the modality gap. As shown in Figure 19, after applying our unitary transformation, image and text embeddings of the same semantic concepts are well-aligned in the shared embedding space while maintaining their internal structure.

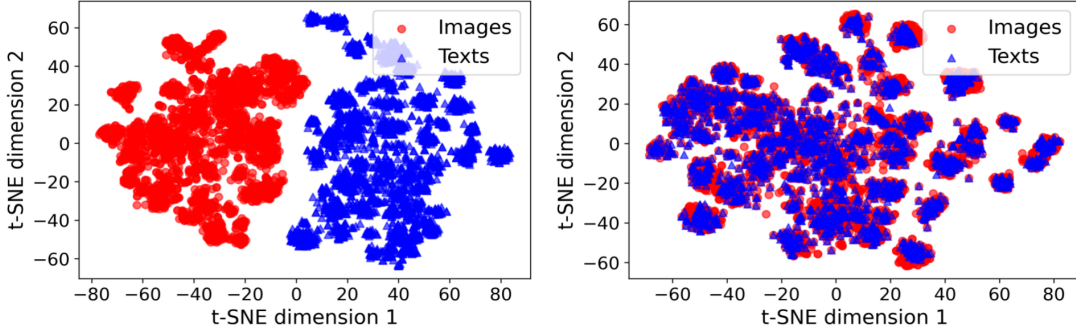


Figure 19: t-SNE visualization of image and text embeddings on MS COCO dataset. **Left:** image embeddings (red circle) and text embeddings (blue triangular) form separate clusters due to modality gap. **Right:** After applying our unitary transformation to text embeddings, the two modalities align well in the embedding space while preserving their internal structure, enabling effective cross-modal clustering.

In the following figures, we showcase the clustering results on the COCO dataset after applying our transformation. This transformation aligns the embedding spaces, enhancing the clustering of semantically similar image-text pairs.

³We emphasize that this step is limited to the cross-modal clustering setting and was not used in any of the other experiments in the paper.



Large commercial cargo **plane** sits on tarmac next to radar equipment.

A side view of a **plane** flying in the clear blue sky.

A white **airplane** parked on a runway under a blue sky.

An **airplane** leaving a trail in the sky.

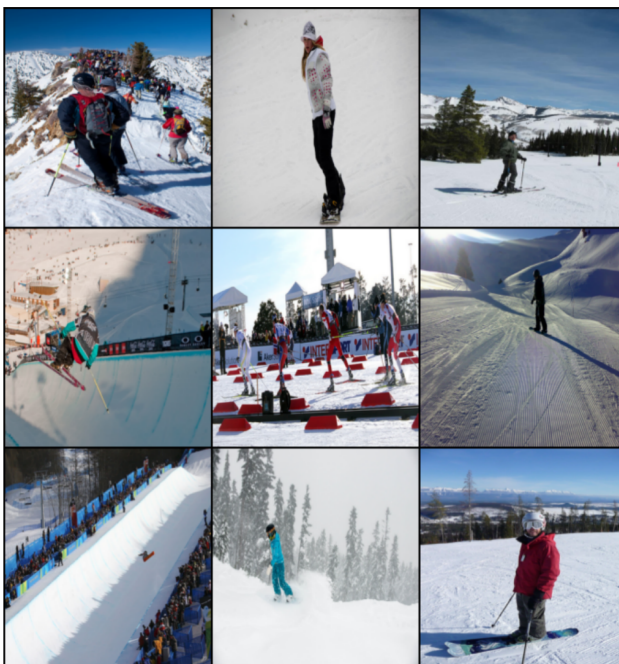
Two **planes** in runway and another plane in the sky.

Some people in high visibility **jackets** putting suitcases onto a conveyer belt from a container.

A black and white image of a **shipyard** with some boats.
A boat traveling into a water filled tunnel.

A couple of **airplanes** flying under a cloudy sky.

Figure 20: Post-transformation alignment of image and text embeddings on the COCO dataset. The cluster samples shown are all related to planes, demonstrating complete alignment of visual and textual data.



A man on his **skis** on a snowy slope.

A person **skiing** through a snowy forest with tall trees.

A man is in his **skiing** gear, while posing for the camera.

a group of beginner snow **skiers** having class

there is a male **skier** going down a hill

a group of people riding **skis** on a snowy surface

A woman with a snowboard with a man standing next to her on a **ski** slope.

a person standing snow wearing a snow suit and **skis**

A woman posing for the camera standing on **skis**.

Figure 21: Post-transformation alignment of image and text embeddings on the COCO dataset. The cluster samples shown are all related to ski, demonstrating complete alignment of visual and textual data.



a person parachute **surfing** on a large body of water

There is a blurry photo of a **surfer** walking out of the water

a man riding a blue **surfboard** on top of a wave.

A man is **surfing** on a wave in the ocean.

A kid on a **surboard** riding a little wave

A man in a wet suit riding a wave on a **surfboard**.

a man wearing a wet suit riding the **wave**

A man and a dog riding a **surf board** in the water .

A man on a blue raft attempting to catch a ride on a large **wave**.

Figure 22: Post-transformation alignment of image and text embeddings on the COCO dataset. The cluster samples shown are all related to surf, demonstrating complete alignment of visual and textual data.



A man standing on a clay **tennis** court with a racket.

A beautiful young woman swinging a tennis racket on a **tennis** court.

A woman prepares to serve a **tennis** ball.

A man holding a **tennis** racket in the air

A man hitting a **tennis** ball with a tennis racket.

Venus Williams is swinging at the **tennis** ball at the game.

A young boy holding a **tennis** racket on a court.

A man holding a **tennis** racket on top of a tennis court.

A woman holding a **tennis** racket while wearing glasses.

Figure 23: Post-transformation alignment of image and text embeddings on the COCO dataset. The cluster samples shown are all related to tennis, demonstrating complete alignment of visual and textual data.

C.3 Text Linear Probe Results

To validate the effectiveness of RP-KrossFuse for improving the uni-modal performance, we also conducted linear probe experiments in text domain. We used the SentEval toolkit [12] to evaluate the RP-KrossFuse embeddings compared to CLIP, Sroberta and other baselines on the following NLP classification benchmarks: MR [58], CR [24], SUBJ [57], MPQA [76], SST2 [68], TREC [72], and MRPC [14]. The linear probe results are provided in the Table 5. The results show that RP-KrossFuse significantly enhances CLIP’s text understanding capabilities, achieving an average improvement of 6% over CLIP itself. In comparison to the baseline, our method demonstrates superior performance across all evaluated tasks.

Table 5: Linear probe evaluation of frozen features of variants of CLIP, Sroberta, RP-KrossFuse and three baselines using the SentEval toolkit on text benchmarks. Test accuracy (%) are based on a 5-fold cross-validation.

Embedding	Arch	Fused	MR	CR	SUBJ	MPQA	SST2	TREC	MRPC	Avg
CLIP [60]	ViT-B/32	✗	75.8	83.1	92.5	86.4	82.0	83.0	70.1	81.8
	ViT-L/14	✗	78.1	85.3	93.8	87.0	83.9	86.4	67.7	83.2
KPoMRP	ViT-B/32	✓	73.8	78.5	86.6	82.1	80.3	74.8	70.8	78.1
	ViT-L/14	✓	72.5	80.3	86.0	83.1	74.0	78.2	68.1	77.5
GATE [66]	ViT-B/32	✓	84.8	87.2	94.4	88.5	91.8	89.3	65.5	85.9
	ViT-L/14	✓	84.3	87.8	94.5	88.3	91.4	89.3	67.6	86.2
ATTN [84]	ViT-B/32	✓	85.7	86.3	93.4	88.8	91.9	88.5	66.2	85.8
	ViT-L/14	✓	85.7	85.9	94.3	87.8	92.4	86.5	65.9	85.5
RP-KrossFuse	ViT-B/32	✓	85.8	88.7	94.4	89.1	89.7	95.0	73.6	88.0
	ViT-L/14	✓	86.0	88.1	94.8	89.3	89.8	95.2	73.6	88.1
SRoBERTa [62]	TF-L24	✗	85.1	86.8	93.7	87.7	89.1	93.2	68.1	86.2

C.4 Zero Shot Image-to-text and Text-to-image Retrievals

To evaluate the cross modal alignment ability of ours RP-KrossFuse method, we conduct zero-shot image-to-text and text-to-image retrieval experiments on MSCOCO [42] and Flickr30k [77]. In our experiments, we utilize several variants of CLIP, including the base, large, and large with 336 pixel models, as well as the large and huge versions of OpenCLIP, to ensure a comprehensive comparison. The results in table 6 shows that the differences between CLIP and RP-KrossFuse are mostly below 1%, suggesting that RP-KrossFuse maintains strong zero shot cross-modal alignment, with retrieval performance comparable to CLIP.

C.5 Improvement of Image Representation vs Alignment of Zero Shot Classification

We performed linear probe and zero-shot classification tasks on multiple datasets covering different visual recognition settings: ImageNet [13], CIFAR-10, CIFAR-100 [35], Caltech101 [16], Food101 [9], Oxford Flowers [51], Oxford-IIIT Pet [59], and DTD [11]. The detailed results are shown in Table 7. As shown in Table 7(a), RP-KrossFuse improves the averaged linear probe accuracy over several image benchmarks from 83.3% to 91.2%. A more detailed breakdown of per-dataset image improvement is visualized in Figure 24(left) where we observe more than 10% increases on 4 out of 8 datasets. Notably, on the ImageNet dataset, RP-KrossFuse irpved over CLIP from 73.2% to 84.1%, suggesting a considerable improvement in the image representation alignment with the actual label. On the other hand, the averaged zero-shot accuracy of RP-KrossFuse drops by 0.5% compared to CLIP(see Table 7(a)). This is indeed expected because we involved uni-modal expert’s embedding which is not aligned with the other modality whereas CLIP is trained to align image and text explicitly. Also, we note that most lower scores are underperforming by at most 1% (on 7 out of 8 datasets), which are relatively outweighed by the gains in uni-modal representations shown in Figure 24 (left). For example, for ImageNet where RP-KrossFuse suffers the gap of -1.57%, the boost in uni-modal representation of 10. 90% is the second highest accuracy over the datasets.

Table 6: Zero-shot image-text retrieval performance on Flickr30K and MSCOCO. We report Recall@K (%) for Image→Text and Text→Image retrieval. Fuse denotes our RP-KrossFuse fusion method. Superscripts denote the backbone variant: B = ViT-B/32, L = ViT-L/14, L+ = ViT-L/14@336px, H = ViT-H/14.

Method	Flickr30K						MSCOCO					
	Image→Text			Text→Image			Image→Text			Text→Image		
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
CLIP ^B	76.9	94.3	97.8	57.9	82.9	89.0	48.4	73.8	81.6	29.8	54.0	65.0
Fuse ^B	75.7	94.6	97.7	58.0	82.5	89.0	48.9	72.7	81.3	29.1	53.3	64.4
CLIP ^L	85.9	97.3	99.2	64.5	87.1	91.9	56.9	79.6	86.7	35.7	60.4	70.5
Fuse ^L	85.0	97.0	99.0	64.6	86.6	91.9	56.5	79.0	86.2	35.7	60.3	70.2
CLIP ^{L+}	87.7	98.5	99.4	66.8	88.9	93.3	57.3	80.2	87.5	35.9	60.7	70.7
Fuse ^{L+}	87.2	98.2	99.4	67.0	88.5	93.0	57.9	80.3	87.0	35.8	60.3	70.4
OpenCLIP ^L	89.0	98.5	99.3	74.9	92.4	95.6	61.8	83.6	89.9	45.5	70.4	79.0
Fuse ^{OpenL}	89.0	98.4	99.4	74.6	92.5	95.6	62.3	83.8	90.0	45.2	70.2	78.8
OpenCLIP ^H	90.7	99.2	99.7	77.6	94.2	96.6	66.1	86.5	91.9	48.5	72.8	81.0
Fuse ^{OpenH}	91.0	99.3	99.8	77.5	94.0	96.7	66.3	86.5	92.0	48.4	72.7	81.1

Table 7: Comparison of CLIP and RP-KrossFuse in the linear probe and zero-shot classification setting. LP: linear probe. ZS: zero shot.

(a) Average.			(b) ImageNet.			(c) CIFAR-10.		
	LP	ZS		LP	ZS		LP	ZS
CLIP	83.3	69.4	CLIP	73.2	57.9	CLIP	95.0	88.8
RP-KrossFuse	91.2	68.9	RP-KrossFuse	84.1	56.3	RP-KrossFuse	98.6	88.4
Δ	+7.9	-0.5	Δ	+10.9	-1.6	Δ	+3.6	-0.4
(d) CIFAR-100.			(e) Caltech101.			(f) Food101.		
	LP	ZS		LP	ZS		LP	ZS
CLIP	80.0	61.7	CLIP	92.6	85.3	CLIP	87.3	79.2
RP-KrossFuse	90.5	61.0	RP-KrossFuse	95.6	85.2	RP-KrossFuse	90.9	78.5
Δ	+10.5	-0.7	Δ	+3.0	-0.1	Δ	+3.6	-0.7
(g) Oxford Flowers.			(h) Oxford-IIIT Pet.			(i) DTD.		
	LP	ZS		LP	ZS		LP	ZS
CLIP	84.9	63.5	CLIP	87.9	75.5	CLIP	65.3	43.0
RP-KrossFuse	99.7	62.8	RP-KrossFuse	95.3	75.7	RP-KrossFuse	75.2	43.3
Δ	+14.8	-0.7	Δ	+7.4	+0.2	Δ	+9.9	+0.3

C.6 Analysis of Ablation Studies

To test the effect of the different RP-KrossFuse components, we evaluated the classification accuracy on ImageNet and the average accuracy across seven NLP benchmarks in SentEval.

Effect of Image Expert Embeddings. To assess the impact of incorporating additional image expert embeddings on the performance of the image modality, we conducted a linear probe on ImageNet by fusing the CLIP embedding with UniCom [3] as an alternative image expert. As illustrated in Figure 25(a), integrating UniCom with CLIP increases the accuracy from 73.2% to 79.17%, demonstrating nearly 6% improvement over using CLIP alone. Furthermore, fusing CLIP with DINOv2 yields even higher accuracy than with UniCom, indicating that the effectiveness of RP-KrossFuse is closely related to the quality of the expert embeddings employed. Specifically,

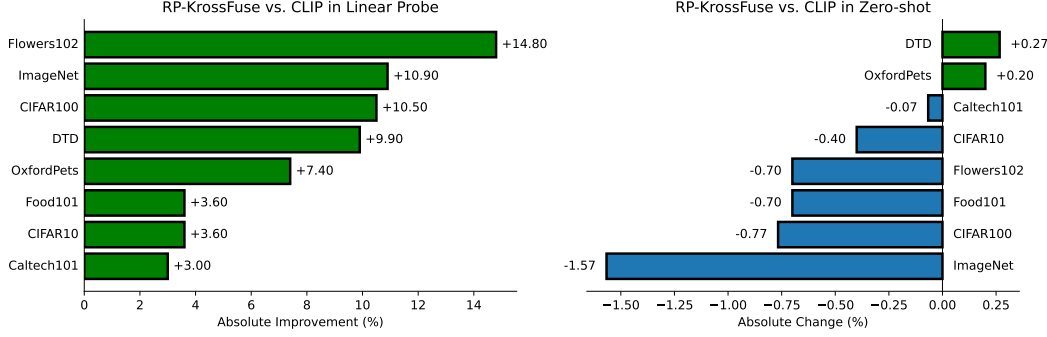


Figure 24: Comprehensive comparisons of RP-KrossFuse and CLIP in the linear probe and zero shot classification settings. (Left) RP-KrossFuse is able to gain consistent improvements over CLIP in linear probe on all datasets. (Right) RP-KrossFuse’s declines in zero shot accuracy are mostly under 1%, which are far outweighed by the gains in linear probe.

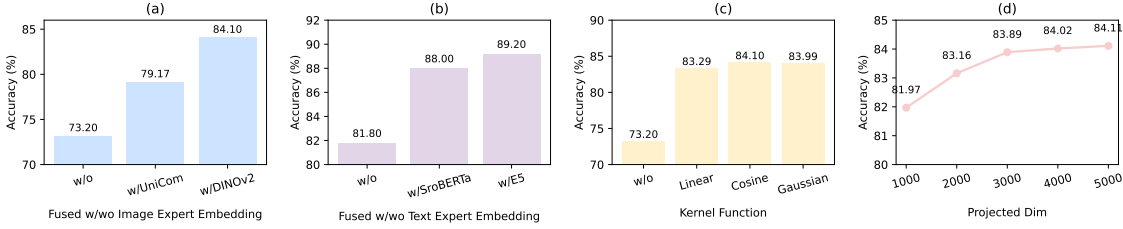


Figure 25: Ablation Studies. (a) (b) Effect of fusing different image and text expert embeddings. (c) Effect of kernel function. (d) Effect of random projected dimension.

since DINOv2 provides stronger representations than UniCom, the overall performance gain is more pronounced when DINOv2 is used as the expert embedding.

Effect of Text Expert Embeddings. To evaluate the generalizability of our approach with different text experts, we replace RoBERTa with E5 in the RP-KrossFuse framework and conduct linear probe experiments using the SentEval toolkit on seven NLP benchmarks: MR [58], CR [24], SUBJ [57], MPQA [76], SST2 [68], TREC [72], and MRPC [14]. As shown in Table 8, incorporating either RoBERTa or E5 as an additional text expert consistently improves the performance over the original CLIP text encoder, with E5 yielding an average accuracy gain of 7.4%. These results demonstrate that RP-KrossFuse can effectively leverage various strong text experts to enhance text representation.

Table 8: Linear probe evaluation of CLIP, E5, SRoBERTa and RP-KrossFuse using the SentEval toolkit on text benchmarks. Test accuracy (%) are based on a 5-fold cross-validation. CLIP+E5 and CLIP+SRoBERTa are RP-KrossFuse methods with different text embeddings.

Embedding	MR	CR	SUBJ	MPQA	SST2	TREC	MRPC	Avg
CLIP	75.8	83.1	92.5	86.4	82.0	83.0	70.1	81.8
E5	87.1	91.1	94.7	90.2	92.0	95.7	73.1	89.1
CLIP+E5	86.8	91.3	94.3	90.5	91.9	95.6	74.1	89.2
SRoBERTa	85.1	86.8	93.7	87.7	89.1	93.2	68.1	86.2
CLIP+SRoBERTa	85.8	88.7	94.4	89.1	89.7	95.0	73.6	88.0

Effect of Cross-modal Embeddings. To assess the generalization capability of RP-KrossFuse in enhancing unimodal performance while maintaining cross-modal alignment, we further conducted linear probe and zero-shot experiments on the ImageNet dataset using several multimodal embedding

baselines. As shown in Table 9, RP-KrossFuse consistently improves linear probe accuracy across different backbones, indicating stronger unimodal representation, while maintaining comparable zero-shot performance, demonstrating preserved cross-modal alignment. These results highlight the generality and effectiveness of RP-KrossFuse when applied to various cross-modal embedding architectures.

Table 9: Comparison of linear probe and zero-shot accuracy (%) between baseline cross-modal embeddings and their RP-KrossFuse counterparts on ImageNet.

Embedding Model	Linear Probe		Zero-Shot	
	Baseline	RP-KrossFuse	Baseline	RP-KrossFuse
SigLIP [80]	81.7	83.7	73.1	72.8
MobileCLIP [70]	82.3	84.5	71.1	70.9
OpenVision [40]	79.3	84.2	65.9	65.1

Effect of Kernel Function. To validate the role of kernel function in RP-KrossFuse framework, we conducted linear probe experiments among different kernel functions. As shown in Figure 25(c), those kernel-based fusions can lead to almost 10% classification accuracy improvements on ImageNet dataset, with the Cosine and RBF kernels yielding slightly higher gains than the linear kernel. In the main text, we use cosine kernel for classification and RBF kernel for clustering.

Effect of Random Projected Dimension. We investigate how the choice of random projection dimension affects the performance of RP-KrossFuse. As shown in Figure 25(d), increasing the projection dimension leads to a steady improvement in classification accuracy, which gradually saturates as the dimension becomes larger. Notably, when the dimension reaches around 3000, the performance of RP-KrossFuse closely approaches that of the KrossFuse, indicating that a sufficiently high projection dimension can effectively preserve the information required for optimal fusion.

Effect of Hyperparameter C . The constant C in Equation (3) balances the influence of uni-modal embeddings on the fused shared-modality kernel similarity and maintains balance between similarity scores for shared and non-shared modalities. Specifically, when considering images and text as shared and non-shared modalities, a smaller C increases the weight of the image uni-modal embedding, but reduces the balance between (image, image), (image, text), and (text, text) similarity scores. To empirically validate this effect, we evaluated the cosine similarity distributions of image-text pairs on the MSCOCO validation set under different values of C . As shown in Figure 26, smaller values of C lead to a slightly larger separation between the similarity curves of CLIP and RP-KrossFuse, whereas larger C results in more overlapping curves, indicating improved cross-modal alignment.

C.7 Computational Efficiency of RP-KrossFuse

We provide additional analysis of the computational efficiency of our proposed RP-KrossFuse scheme. Theoretically, RP-KrossFuse reduces the computational complexity from $\mathcal{O}(d_1 d_2 l)$ (naive random projection) to $\mathcal{O}((d_1 + d_2)l)$ by leveraging structured random projections, which enables more scalable multimodal fusion. Regarding inference efficiency, it is important to note that, assuming sufficient memory, our method allows for the two embedding models to operate in parallel, thereby avoiding increased inference latency. To support this, we conducted benchmarking experiments on two NVIDIA RTX 3090 GPUs with a batch size of 128. The original KrossFuse pipeline required 491.3 ms per batch and 18.406 GB of peak memory, while RP-KrossFuse (with $l = 5000$) achieved comparable performance with just 413.1 ms per batch and a significantly reduced memory footprint of 3.39 GB. Notably, the fusion step’s overhead was reduced from 22.0 ms to only 0.14 ms. For reference, the baseline CLIP and DINOv2 models individually required 73.6 ms and 392.1 ms per batch, respectively.

C.8 Concatenation-Only Baseline

We additionally analyze a concatenation-only baseline to better understand the advantages of the proposed RP-KrossFuse fusion strategy. From a theoretical perspective, concatenating two embeddings corresponds to defining a kernel similarity function that is the *sum* of the individual kernels. This formulation implies that two samples will have high similarity in the fused space only if both original

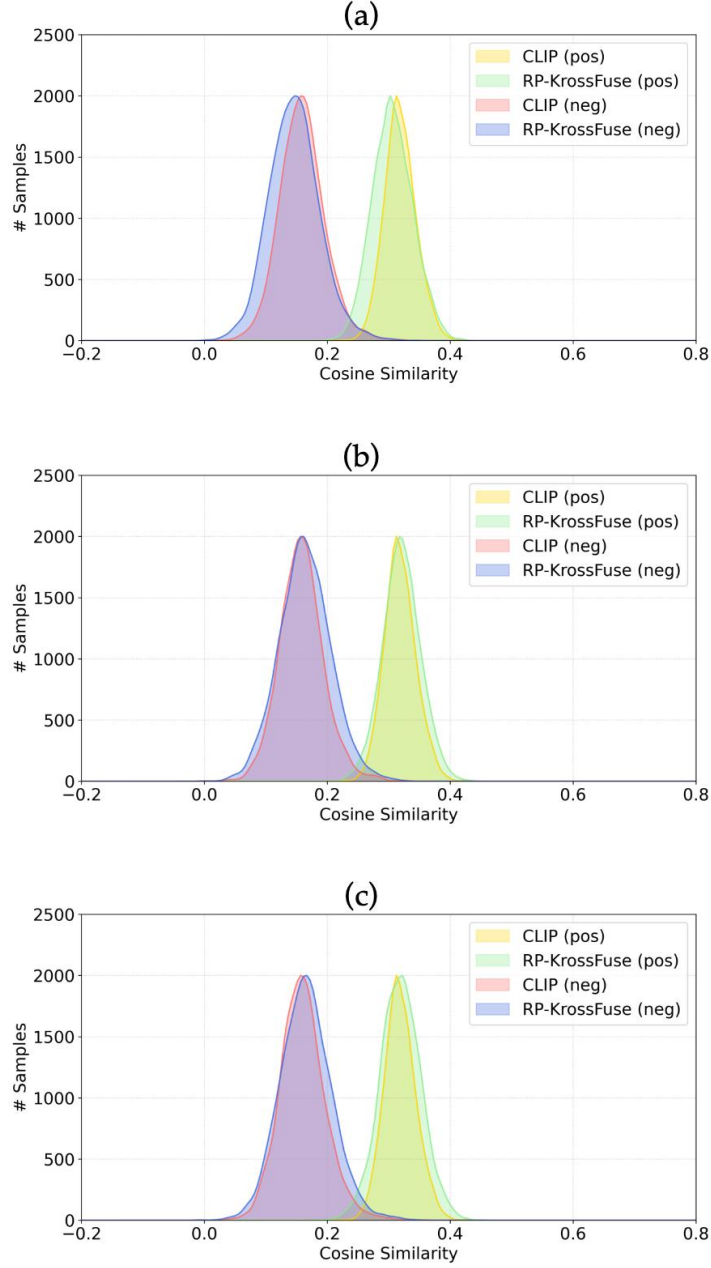


Figure 26: The cosine similarity distributions of positive/negative image text pairs on MSCOCO dataset across various values of hyperparameter C . (a) $C = 0.1$. (b) $C = 0.5$. (c) $C = 1$.

Table 10: Benchmarking of inference and fusion efficiency.

Method	Time/Batch (ms)	Peak Memory (GB)
CLIP	73.6	2.077
DINOv2	392.1	3.199
KrossFuse Pipeline	491.3	18.406
KrossFuse (Overhead)	22.0	17.203
Parallel RP-KrossFuse Pipeline	413.1	3.387
RP-KrossFuse (Overhead)	0.14	1.729

embeddings assign them high similarity, effectively modeling the *intersection* of the semantic concepts captured by each modality. In contrast, the Kronecker-based fusion employed in RP-KrossFuse leads to a *product* of kernel similarities, which enables distinguishing two samples as long as either embedding distinguishes them, thereby modeling the *union* of their semantic representations.

In supervised classification settings, the concatenation baseline is expected to perform comparably to the stronger of the individual embeddings. This is because the concatenated feature vector preserves the full representational capacity of both embeddings, while the limited VC dimension of the linear probe mitigates overfitting. However, in unsupervised or weakly supervised scenarios such as clustering, the product-based Kronecker fusion can better capture complementary cross-modal structures, leading to improved representation quality and broader concept coverage.

Empirically, we compared the concatenation baseline with RP-KrossFuse across multiple tasks. For supervised ImageNet linear probing, both methods achieve similar performance (concatenation: 83.6%, RP-KrossFuse: 84.1%), which aligns with theoretical expectations. In multimodal classification tasks, RP-KrossFuse consistently outperforms the concatenation baseline on the MVSA dataset [52], achieving 74.3% vs. 71.3% on MVSA-Single and 66.6% vs. 63.7% on MVSA-Multiple. The largest gains are observed in unsupervised clustering tasks, where RP-KrossFuse shows clear improvements across datasets, as summarized in Table 11.

Table 11: Comparison between the concatenation-only baseline and RP-KrossFuse on unsupervised clustering tasks, evaluated using NMI, AMI, and ARI metrics (%).

Method	Metric	Flowers102	DTD	ImageNet-Dogs	GTSRB	Typo-Attacked IN
Concatenation	NMI	98.4	60.5	76.3	43.8	44.4
RP-KrossFuse	NMI	99.1	62.9	88.3	50.0	87.4
Concatenation	AMI	97.9	57.9	76.1	40.2	43.9
RP-KrossFuse	AMI	98.8	60.4	88.2	46.7	87.3
Concatenation	ARI	93.9	35.3	64.9	14.1	28.4
RP-KrossFuse	ARI	97.0	36.4	86.3	19.5	79.6

C.9 Application of RP-KrossFuse in Text-to-Image Diffusion Models

To further demonstrate the practical utility of the proposed RP-KrossFuse embedding fusion, we apply it to conditional image generation with text-to-image diffusion models. Specifically, we adapt the Vendi Score Guidance (VSG) framework proposed by Askari *et al.* [4], which guides the reverse diffusion process using the Vendi diversity score of the generated samples. In the original implementation, CLIP embeddings were used to compute the Vendi diversity score. We replace this embedding with the Kronecker-fused representation of DINOv2 and CLIP, thereby enriching the diversity guidance with complementary visual and multimodal semantics.

Experiments were conducted on the ImageNet dataset using a class-conditional Diffusion Transformer (DiT-XL/2) as the backbone. Two guidance configurations were compared: (1) the original CLIP-based contextualized Vendi Score Guidance (c-VSG), and (2) our KrossFuse-based version, which employs the fused CLIP and DINOv2 embeddings. As reported in Table 12, incorporating RP-KrossFuse improves both diversity and fidelity of the generated samples. Specifically, the diversity metrics—Recall [36] and Coverage [49]—increase notably, while the quality metrics—Precision [36] and Density [49]—also improve over the CLIP-only baseline.

Table 12: Comparison of Vendi Score diversity guidance methods for text-to-image diffusion on ImageNet using DiT-XL/2. Metrics follow [36, 49].

Diversity Guidance Method	Precision	Recall	Density	Coverage
c-VSG Guidance (CLIP)	0.913	0.413	1.206	0.552
KrossFuse (CLIP + DINOv2)	0.932	0.484	1.252	0.613

NeurIPS Paper Checklist

1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: [\[Yes\]](#)

Justification: The abstract and introduction clearly summarize our proposed method and its contributions. These claims are directly supported by our experimental findings across multiple benchmarks.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [\[Yes\]](#)

Justification: Our paper includes a "Conclusion and Limitations" subsection in the main text.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory assumptions and proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: Our paper provide complete proofs in the section A. of appendix.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. Experimental result reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: Our paper ensures full reproducibility. The experiments details are provided in the appendix.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
 - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
 - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
 - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
 - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We submitted the code as the part of the supplementary materials.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. Experimental setting/details

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: Details are given in Sec. B. of the appendix.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. Experiment statistical significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: We report the average performance over multiple runs, but do not include error bars or statistical significance tests. The observed variance across runs was small, so we focused on reporting mean results for clarity.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments compute resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: All experiments are run on two RTX-4090 GPUs.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. Code of ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [Yes]

Justification: The paper conforms with the NeurIPS Code of Ethics and does not pose any potential harm.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. Broader impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [NA]

Justification: We do not foresee any negative societal impacts beyond those generally associated with large-scale representation learning.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: The paper poses no such risks.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: We cite all models and datasets we used in the paper.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification: The paper does not release new assets.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. Declaration of LLM usage

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>) for what should or should not be described.